

# RNN-GAN을 이용한 코드 기반의 단계적 트로트 음악 생성 기법

## Chord-based stepwise Korean Trot music generation technique using RNN-GAN

황서림,<sup>1</sup> 박영철<sup>†</sup>

(Seo-Rim Hwang<sup>1</sup> and Young-Cheol Park<sup>1†</sup>)

<sup>1</sup>연세대학교 컴퓨터정보통신공학부

(Received October 1, 2020; revised November 2, 2020; accepted November 11, 2020)

**초 록:** 본 논문은 순환 신경망(Recurrent Neural Network, RNN)으로 구성된 적대적 생성 신경망(Generative Adversarial Network, GAN) 모델을 사용하여 자동으로 트로트 음악을 생성하는 음악생성 기법을 제안한다. 제안된 방법은 음악의 뼈대를 담당하는 코드를 만들고, 만들어진 코드 열을 기반으로 멜로디와 베이스(bass)를 단계적으로 생성한 뒤, 해당 코드에 붙임으로써 구조화된 음악을 완성하는 방법을 사용한다. 또한인트로나 벨스, 코러스 등과 같이 일정 구간으로 나뉘어 구조가 반복되는 트로트 가요의 특징을 적용하여 벨스의 코드 진행으로부터 새로운 코러스 코드 진행을 만들어내고, 다시 해당 코드로부터 멜로디와 베이스를 단계적으로 생성하여 초기에 만들어진 트로트의 길이를 확장한다. 주관적 평가와 객관적 평가방법을 사용하여 생성된 음악의 품질을 측정하였으며, 기존의 트로트가 갖고 있는 음악적 특성과 유사한 음악을 생성함으로 확인하였다.

**핵심용어:** 음악 생성, 코드 기반 생성, 적대적 생성 신경망, Bidirectional Long Short-Term Memory (BLSTM)

**ABSTRACT:** This paper proposes a music generation technique that automatically generates trot music using a Generative Adversarial Network (GAN) model composed of a Recurrent Neural Network (RNN). The proposed method uses a method of creating a chord as a skeleton of the music, creating a melody and bass in stages based on the chord progression made, and attaching it to the corresponding chord to complete the structured piece. Also, a new chorus chord progression is created from the verse chord progression by applying the characteristics of a trot song that repeats the structure divided into an individual section, such as intro, verse, and chorus. And it extends the length of the created trot. The quality of the generated music was specified using subjective evaluation and objective evaluation methods. It was confirmed that the generated music has similar characteristics to the existing trot.

**Keywords:** Music generation, Chord-based generation, Generative Adversarial Network (GAN), Bidirectional Long Short-Term Memory (BLSTM)

**PACS numbers:** 43.75.St, 43.75.Zz

### 1. 서 론

음악은 대표적인 시퀀스 데이터로서 특유의 주관적 요소를 포함하고 있어서 자동 생성이 어려운 분

야 중 하나이다. 최근 딥러닝 기술의 발전과 함께 다양한 신경망을 사용하여 자연스러운 음악을 만들어 내기 위한 연구가 활발하게 진행되고 있다.<sup>[1-5]</sup> 특히 적대적 생성 신경망(Generative Adversarial Network,

<sup>†</sup>Corresponding author: Young-Cheol Park (young00@yon-sei.ac.kr)

Division of Computer and Telecommunication Engineering, Yonsei University, Chang jo room 269, 1 Yonsei-dae-gil, Wonju, Gangwon-do 26493, Republic of Korea

(Tel: 82-33-2744, Fax: 82-33-763-4323)



Copyright©2020 The Acoustical Society of Korea. This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

GAN)을 사용한 음악 생성 기법은 음악 자동 생성의 가능성을 높이고 있다.<sup>[2-4]</sup> Dong *et al.*<sup>[4]</sup> 제안한 MuseGAN은 여러 개로 구조화된 생성자와 판별자를 사용하여 다양한 악기가 연주되는 멀티 트랙 음악을 만들 수 있음을 보였다. 그러나 음악 데이터의 복잡성 때문에 기존 음악 생성 연구들은 길이가 긴 음악은 생성하는데 어려움이 있다. 한편, Wu *et al.*<sup>[11]</sup>이 제안한 논문<sup>[11]</sup>에서는 입력 데이터를 세분화하고 순환신경망(Recurrent Neural Network, RNN)을 다층으로 쌓아 적절한 길이의 음악을 만들 수 있음을 보였다. 그러나 멜로디만을 생성하였기 때문에 음악이 단조롭다는 단점이 있다.

본 논문에서는 순환신경망으로 구성된 GAN 모델을 사용하여 트로트 음악을 자동 생성하는 기법을 제안한다. 이를 위해 트로트의 코드와 음악적 구성 요소 간 관계를 고려한 코드 기반의 작곡 방식을 사용한다. 제안된 방법에서는 트로트 음악의 악보로부터 코드와 멜로디, 베이스로 분리하고, 화성학을 기반으로 하여 라벨링함으로써 코드 기반의 음악 생성을 구현한다. 이를 통해 상대적으로 길이가 긴 멀티 트랙 음악을 생성할 수 있다. 코드 생성을 위해 RNN-GAN을 사용하며, 조건부 RNN-GAN 모델<sup>[5,6]</sup>을 사용하여 코드로부터 멜로디와 베이스를 생성한다.

본 논문의 구성은 다음과 같다. II장에서 트로트 음악 생성을 위한 데이터 표현법에 대해 설명한다. III장에서는 코드 기반의 트로트 생성 모델과 알고리즘을 설명하고, IV장에서는 생성된 트로트 음악에 대한 분석과 성능을 평가한다. 마지막으로 V장에서 결론을 맺는다.

## II. 트로트 음악 생성을 위한 데이터 표현

트로트 음악 생성기를 훈련하기 위해서는 기존의 트로트 음악의 특성을 데이터화할 수 있는 표현 형식이 필요하다. 기존 음악 생성 기법들은 코드, 멜로디 등을 분리하지 않고 생성하거나 멜로디만 단일 생성하는 방법<sup>[11]</sup>을 사용하였다. 그러나 이러한 접근법은 데이터가 가진 방대한 정보량 때문에 모델이 제대로 학습을 못 하거나 음악의 구조를 단조롭게

만드는 어려움이 있다. 본 논문에서는 곡마다 붙여진 스케일을 기반으로 음을 라벨링하는 방법을 사용하여 음악 데이터의 구조를 단순화하였다.

### 2.1 스케일 기반 라벨링

스케일이란 악곡을 구성하는 음을 나타내는 지표를 의미한다. 즉, 곡의 스케일을 알면 해당 곡이 진행될 수 있는 음의 구간을 알 수 있다. 가요 악보에는 해당 악보의 스케일을 표현하는 키(key)가 존재한다. 작곡가의 주관적인 판단에 의해 키의 스케일을 따르지 않는 예외적인 코드나 음들이 사용되기도 하지만, 일반적으로 곡은 키가 제시해주는 스케일 내에서 진행된다. 멜로디와 베이스 또한 코드와 키의 스케일 내에서 선택된다.

본 논문에서는 스케일에 따른 코드와 멜로디, 베이스 간의 연관성에 집중하였고, 코드와 키 스케일을 병합한 범위 내에서 멜로디와 베이스를 라벨링함으로써 음악 데이터 표현을 간소화하였다. 이와 같은 방식은 음악 데이터 구조를 간략화 할 수 있는 장점이 있다. 그리고 같은 라벨링 값을 가지더라도 주어지는 코드의 값에 따라 실제 음이 달라지기 때문에 데이터 구조의 간략화에도 불구하고 다양한 멜로디 조합을 표현할 수 있다. 또한, 곡의 스케일에 기반

Cm key = C D D# F G G# A#  
Ab chord = G# C D#

Ab Chord

Melody

Bass

Key	Chord	CR	CO	MR	MO	ML	BR	BO	BL
Cm	Ab	0.5	2	0.5	2	1	4	3	2
	Ab	0.5	2	0.5	0	0	0	0	0
	Ab	0.5	2	0.5	3	3	0	0	0
	Ab	0.5	0	0.5	2	3	0	0	0
	Ab	0.5	0	1.25	0	0	0	0	0
	Ab	0.5	0	0.75	2	5	0	0	0
	Ab	0.5	2	0	0	0	0	0	0
	Ab	0.5	2	0	0	0	0	0	0

Fig. 1. An example of preprocessed data from a bar (CR: chord rhythm, CO: chord octave, MR: melody rhythm, MO: melody octave, ML: melody label, BR: bass rhythm, BO: bass octave, BL: bass label).

하여 라벨링함으로써 최종적으로 생성된 결과가 화학적으로 안정된 결과를 기대할 수 있다.

이와 같은 접근법에 따라, 먼저 코드를 코드명과 박자, 옥타브로 3차원 데이터로 구성하였다. 멜로디와 베이스 또한, 각각 박자와 옥타브, 그리고 어떤 제약조건 내에서 연주되었는지에 대한 라벨링 값을 포함한 3차원 데이터로 구성하였다. Fig. 1은 이와 같은 스케일 기반 라벨링 방식을 사용하여 1마디의 3중악 보로부터 추출한 데이터를 예시하여 보여주고 있다.

## 2.2 트로트의 음악적 특징을 사용한 곡 생성

트로트는 일반적으로 노래의 도입부를 나타내는 인트로, 1절, 1절과 2절을 이어주는 브릿지, 2절, 그리고 노래의 마무리를 장식하는 아웃트로로 구성되며 각 절은 주 멜로디 파트에 해당하는 벌스와 후렴구에 해당하는 코러스로 이루어진다. 특정 박자로 이루어지는 곡의 최소 단위를 마디라고 할 때, 가요는 보통 벌스 16마디, 코러스 16마디 길이로 구성된다. 트로트는 벌스와 코러스 내에서 4마디 혹은 8마디씩을 기준으로 'A-A'-B-C', 'A-A-B-B', 'A-A' 등과 같은 반복적인 구조로 구성된다는 특징이 있다. Fig. 2는 코드와 멜로디 정보를 나타내는 트로트 악보에서 벌스 부분 16마디만 발췌한 것으로 8마디를 기준으로 1-1과 1-2, 2-1과 2-2의 코드와 멜로디 모두 유사하게 반복되는 것을 확인할 수 있다.

음악 데이터는 하나의 트랙으로도 방대한 정보량을 가지며 복잡한 구조를 띠기 때문에 멀티트랙 환경의 음악을 생성할 때 입력 데이터를 세분화시켜주는 과정은 중요하다. 본 논문에서는 트로트의 특성

을 분석하여 벌스와 코러스로 나누고, 각각의 벌스와 코러스 또한 4마디 단위로 구분하여 모델에 입력에 넣었으며 2.1절에서 언급한 라벨링 기법을 통하여 데이터를 단순화하였다. 이를 통해 모델 학습 가능성을 높이고, 길이가 상대적으로 긴 멀티트랙 음악 생성이 가능하도록 하였다.

실제 트로트 음악들 중 위에 언급한 전형적 특징에서 벗어나는 경우들이 많이 존재할 수 있다. 그러나 본 논문에서는 음악 생성의 효율성을 위해 위와 같은 트로트의 전형적 특성을 가정하였다. 이 가정을 통해 벌스와 코러스를 구분하고, 벌스를 통해 코러스로 곡을 확장할 수 있도록 데이터를 구성하였다.

## III. 트로트 음악 생성

본 논문에서는 음악 생성의 복잡성을 완화하기 위하여 생성의 단계를 나누어서 점진적으로 음악의 복잡도를 증가시키는 접근법을 사용한다. 이에 따라 먼저, 코드를 생성하고 만들어진 코드를 기반으로 멜로디와 베이스를 덧붙이도록 하였다.

### 3.1 단계적인 트로트 생성 모델

본 논문에서 사용한 트로트 생성 과정은 Fig. 3와 같다. 먼저 악보(sheet music)로부터 벌스와 코러스를 나누고, 각각 전처리 과정(pre processing)에서 2.1절에서 설명한 라벨링 과정을 통해 훈련용 악보로부터 Fig. 1과 같은 형태의 구조화된 음악 데이터를 추출해 낸다. 이때 곡 안의 코드들은 코드 열(chord progression)의 형태로 만들어지며, 멜로디와 베이스로부터 분리된다. 분리된 벌스 코드 열은 RNN-GAN 모델의 입력으로 들어가 16마디의 새로운 벌스 코드 열을 생성한다. RNN-GAN으로부터 생성된 벌스 코드와 사전에 추출된 멜로디와 베이스는 각각 조건부 RNN-GAN 모델의 조건과 입력으로 들어가 16마디의 새로운 벌스의 멜로디와 베이스, 그리고 이후 이어질 코러스 코드 열을 생성하는 데에 사용된다. 새로운 코러스 코드 열이 생성되면, 벌스의 멜로디와 베이스를 생성했던 방법과 동일하게 코러스 코드 열과 사전에 추출된 코러스 멜로디와 베이스를 각각 조건부 RNN-GAN 모델의 조건과 입력으로 넣어 멜

Fig. 2. (Color available online) An example of trot data that repeat based on 8-bar.

티 트랙 코러스를 만든다. 그리고 만들어진 벌스와 코러스를 연결하여 최종 트로트 음악을 완성한다.

RNN-GAN의 경우 랜덤 잡음을 입력으로 받아 새로운 코드 진행을 만든다. 한편 조건부 RNN-GAN은 생성자에 랜덤 잡음 외에도 판별자에게도 해당 조건을 사전에 알려주는 방식을 채택한다.<sup>[6]</sup> Fig. 3에서 조건부 RNN-GAN은 RNN-GAN을 통해 만들어진 코드 진행을 조건으로 받아서 해당 코드에 맞는 멜로디와 베이스 혹은 새로운 코드 진행을 생성한다. 이때 모델 학습에는 코드 진행과 해당 코드 진행에 따른 멜로디, 베이스가 각각 매칭 되어 사용된다. 이와

같은 과정을 통하여 최종적으로 멀티트랙 환경의 음악을 생성한다.

RNN-GAN과 조건부 RNN-GAN은 모두 음악과 같은 시퀀스 데이터 모델에 적합한 *bidirectional LSTM*(BLSTM)을 사용하여 구성하였다. 세부적인 네트워크 구성은 Table 1과 같다.

### 3.2 훈련 데이터 구성

본 논문에서는 코드의 기본이 되는 3도 화음 코드만을 다루었다. 코드명의 경우 다장조는 해당 스케일 내의 코드인 C, Dmin, Emin, F, G, Amin, Bdim을, 단장조는 Cm, Ddim, Eb, Fmin, Gmin, Ab, Bb를 우선으로 두고, 사용된 악보들 내에서 쓰이는 코드를 추가하여 one-hot으로 인코딩한 벡터로 임베딩 하였다. 멜로디와 베이스도 전처리 과정을 통해 얻어진 값을 코드와 같은 방법으로 임베딩 하였다. 모든 악보의 키는 장음계와 단음계로 한정하였으며, 다른 키를 갖는 곡은 다장조 또는 단장조로 조옮김한 뒤에 데이터화하였다. 그리고 Chen *et al.*<sup>[7]</sup>이 사용한 예외처리 방식과 유사하게 C7나 Caug와 같은 코드는 C로 바꾸어 3화음 코드로 조정하였다.

박자는 4분 음표를 1로 계산하여 다른 음표들의 값을 상대적으로 지정하였다. 옥타브는 데이터 셋 내에서 사용된 C0~C4까지 총 35개 음, 즉 5개의 범위로 제한하였다. 이러한 조건은 생성 음악의 다양성을 제한할 수 있으나, 트로트의 음악적 형태가 비교적 단조롭고 반복되는 특징을 가지고 있기 때문에

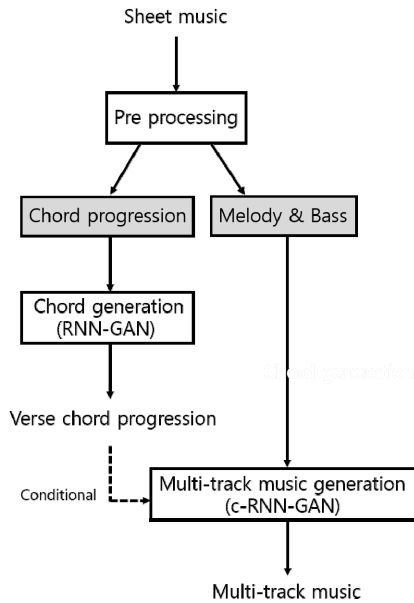


Fig. 3. The architecture of trot generation model.

Table 1. Configuration of the RNN-GAN and conditional RNN-GAN.

	RNN-GAN		c-RNN-GAN	
	Generator	Discriminator	Generator	Discriminator
Input	16*16*3 (random noise)	16*16*3 (data representation)	16*16*3 + 16*16*3 (real data + random noise)	16*16*3 + 16*16*3 (real data + generated data)
Layer 1	256, BLSTM, tanh	256, LSTM, tanh	1024, BLSTM, tanh	1024, BLSTM, tanh
Layer 2	256, BLSTM, tanh	256, LSTM, tanh	1024, BLSTM, tanh	1024, BLSTM, tanh
Layer 3	512, Fully-connected, tanh	512, Fully-connected linear	512, Fully-connected, tanh	512, Fully-connected sigmoid
Layer 4	512, Fully-connected, linear	512, Fully-connected, sigmoid	512, Fully-connected, linear	x
Output	16*16*3 (data representation)	2 (real or fake)	16*16*3 (data representation)	2 (real or fake)

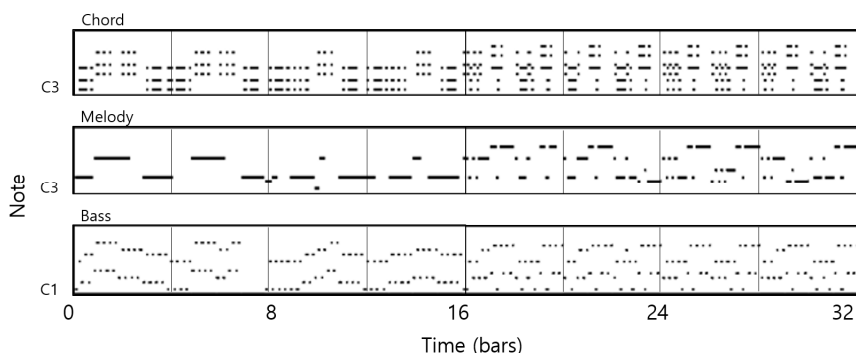


Fig. 4. One sample of the generated trot musics consisting of chords, melody and bass. The sample consists of 16-bar verse and 16-bar chorus and is about 64 seconds long. The left side is the verse and the right side is the chorus.

미치는 영향은 크지 않다.

최종적으로 생성모델 훈련에 실제로 사용된 데이터는 트로트 장르에 해당하는 장음계 29곡, 단음계 악보 117곡으로서 벨스 141개, 코러스 136개이다. 마디로 계산한다면, 총 4,352 마디의 데이터를 사용하였다. 4/4 박자를 따르지 않거나 일반적인 트로트 양식을 따르지 않는 예외적인 악보들은 대상에서 제외하였다.

#### IV. 실험결과 및 평가

제안한 생성 모델을 117곡의 트로트 데이터를 사용하여 훈련하였다. 생성자와 판별자의 학습 비율은 5대 1로 하여 생성자가 5번 훈련될 때 판별자는 1번 학습할 수 있도록 조정하였다. 학습에 사용되는 학습을 또한 생성자는 0.001, 판별자는 0.005로 차이를 두어 생성자와 판별자가 차등적으로 훈련될 수 있도록 하였다. 이를 통해 생성자보다 판별자의 성능이 너무 뛰어나 생성자가 제대로 학습하지 못 하는 경우를 방지하였다.

Fig. 4는 최종 생성된 음악을 피아노 롤 형태로 시각화하여 보여주고 있다. 출력은 벨스와 코러스에 해당하는 코드, 멜로디, 베이스로 총 3개의 트랙으로 구성된다. 코드와 멜로디는 C3 옥타브부터, 베이스는 C1 옥타브부터 2단계의 옥타브 내에서 표현한 것이다. 트로트의 반복적인 특징을 고려하여 생성기 모델 내부의 입출력 데이터는 4마디를 기준으로 분할하여 사용하였기 때문에 합성된 멀티 트랙 음악

또한 4마디를 단위로 반복된다. Fig. 4 왼쪽의 벨스는 오른쪽의 코러스에 비해 더 긴 박자로 구성되어 단조로운 경향을 보인다. 이는 일반적인 벨스와 코러스의 특성 차이가 반영된 것이다.

본 논문에서는 MuseGAN<sup>[4]</sup>에서 사용된 정량적 수치를 사용한 객관적 평가와 주관적 평가 방법을 사용하여 모델의 성능을 평가하였다. 객관적 평가에서는 음악의 구조를 Reference [4]에서 사용한 몇 가지 항목으로 점수화하여, 제안된 모델(proposed)이 합성한 음악과 실제 트로트 음악(train set)의 유사도를 측정하였다. 사용한 평가 항목은 비어있는 마디의 비율을 나타내는 Empty Bar(EB), 0부터 12까지 각 마디 당 사용된 음조 구간의 수를 나타내는 Used Pitch Classes(UPC), 음조 간 거리를 나타내는 Tonal Distance(TD)이다. EB와 UPC 값은 훈련 데이터와 가까울수록 좋으며, TD의 경우 값이 낮을수록 좋다.

비교 결과는 Table 2에 주어져 있다. train set의 경우, 훈련에 사용된 모든 트로트 곡으로부터 측정된 평가 항목을 평균하여 얻었으며, 제안된 모델의 경우, 초기 값을 달리하여 생성한 10곡의 샘플 곡으로부터 측정된 값을 평균하였다. 제안된 모델의 EB, UPC 점수는 코드, 멜로디, 베이스 모두 train set의 EB, UPC 점수와 유사하다. 이는 모델이 잘 학습되어 생성된 데이터가 실제 훈련 데이터와 유사한 음조 구간을 사용하며 적절한 심포 값을 가지는 것을 의미한다. TD 점수는 train set보다 더 좋은 값을 보이는데 생성된 음악의 코드와 멜로디, 베이스 간의 관계가 밀접하게 구성되었음을 알 수 있다.

Table 2. Objective evaluation. Values within the quadrant of Real in EB and UPC are indicated in bold and the smallest TD scores are indicated in bold (C: chord, M: melody, B: bass; EB and UPC values are better as they are closer to origin and lower TD is better).

train set	empty bar ratio (EB; %)			used pitch classes (UPC; %)			tonal distance (TD)		
	C	M	B	C	M	B	C-M	C-B	M-B
	19.22	20.26	3.08	13.47	17.91	23.87	1.38	1.52	1.27
proposed	C	M	B	C	M	B	C-M	C-B	M-B
	<b>27.34</b>	<b>21.88</b>	<b>6.25</b>	<b>13.02</b>	<b>13.54</b>	<b>22.39</b>	<b>0.19</b>	<b>1.26</b>	1.40

Table 3. Subjective evaluation (H: harmonic, R: rhythmic, MS: musically structured, T: trot, Avg: average; 5 is the highest score).

		H	R	MS	T	Avg
non-pro	real	4.75	4.75	4.75	4.83	4.75 / 4.77
	proposed	<b>4.25</b>	<b>4.58</b>	<b>4.02</b>	2.50	<b>4.28</b> / 3.83
	Ref. 1	2.95	3.37	2.38	-	2.90
	Ref. 2	3.13	4.13	3.48	-	3.58
pro	real	3.87	3.74	3.74	3.50	3.78 / 3.71
	proposed	<b>4.00</b>	<b>3.74</b>	<b>3.58</b>	3.37	<b>3.77</b> / 3.67
	Ref. 1	3.84	3.52	3.45	-	3.60
	Ref. 2	3.87	3.66	3.23	-	3.75

한편, EB, UPC, TD는 곡의 장르에 따라 분포가 달라질 수 있기 때문에 다른 음악 생성 기법과의 비교를 위해 사용하는 것은 적절하지 않다. 따라서 제안된 기법과 다른 음악 생성 기법의 비교는 주관적인 평가 방법을 사용하였다.

주관적 평가는 훈련에 사용되지 않은 실제 트로트 음악(real)과 제안된 모델, 2개의 비교 모델을 대상으로 진행하였다. 비교 모델로는 각각 Ref. 1<sup>[1]</sup>과 Ref. 2<sup>[4]</sup>를 사용하였다. 평가는 화성을 잘 지켰는지(H), 리듬이 있는지(R), 음악의 구조화가 잘 되어있는지(MS), 트로트의 음악적 특성이 나타나는지(T)등을 5점 만점으로 측정하도록 하였으며, Ref. 1과 Ref. 2는 트로트를 생성하지 않기 때문에 (T) 항목에 대한 평가는 제외하였다. 평가에는 음악 전공자(pro) 4인과 비전공자(non pro) 6인이 참여하였다. 비전공자는 평가 항목에 대한 사전 교육을 실시하였다. 평가는 평가자가 평가하는 곡들에 대한 정보가 없이 무작위 순서로 제공하여 블라인드로 진행하였다. 평가결과는 Table 3과 같다.

전공자들은 제안된 모델이 Real과 비교하여 구조화(MS)와 트로트 특성(T)이 부족하다고 평가하였다.

이는 데이터 셋의 부족으로 인하여 벌스와 코러스가 뚜렷하게 구분되지 않았거나 벌스와 코러스 내에서 음악의 진행이 너무 단조롭고, 트로트 특성이 잘 반영되지 못했기 때문인 것으로 추정된다. 그러나 그 밖의 항목에서는 real에 비해 좋은 점수를 얻거나 동일한 점수를 얻었으며, 결과적으로 각 항목 평균 점수에서도 0.04점의 작은 차이를 보인다. 또한, 제안된 모델은 전공자와 비전공자 평가 모두 모든 항목에서 Ref. 1과 Ref. 2보다 높은 점수를 보인다.

## V. 결론

음악생성 기법을 대중가요인 트로트 장르에 적용하였다. 화성학에 기반을 둔 함축적인 데이터 표현 방식을 제안하였고, RNN-GAN과 조건부 RNN-GAN을 사용하여 코드로부터 멜로디를 생성하는 단계적인 생성방법을 제안하였다. 제안된 방법은 멀티 트랙 환경에서 적절한 길이의 곡을 효율적으로 생성할 수 있다는 장점이 있으며, 실험을 통해 자연스러운 트로트 음악 생성이 가능함으로 보였다. 향후 데이터의 수를 늘리고 인트로, 브릿지, 아웃트로 등을 추

가적으로 생성하여 연결한다면, 보다 완성도 높은 음악을 생성할 수 있을 것으로 기대한다.

## References

1. J. Wu, C. Hu, Y. Wang, X. Hu, and J. Zhu, "A hierarchical recurrent neural network for symbolic melody generation," *IEEE Trans. on Cybernetics*, **50**, 2749-2757 (2019).
2. Y. Yu and S. Canales, "Conditional lstm-gan for melody generation from lyrics," *arXiv preprint arXiv:1908.05551* (2019).
3. N. Zhang, "Learning adversarial transformer for symbolic music generation," *IEEE Trans. on Neural Networks and Learning Systems*, 1-10 (2020).
4. H.-W. Dong, W.-Y. Hsiao, L.-C. Yang, and Y.-H. Yang, "Musegan: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment," *Proc. the AAAI Conf. on Artificial Intelligence*, 34-41 (2018).
5. S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, **9**, 1735-1780 (1997).
6. M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv:1411.1784* (2014).
7. K. Chen, W. Zhang, S. Dubnov, G. Xia, and W. Li, "The effect of explicit structure encoding of deep neural networks for symbolic music generation," *Proc. MMRP*. 23-24 (2019).

## 저자 약력

### ▶ 황 서 림 (Seo-Rim Hwang)



2017년 3월 ~ 현재: 연세대학교 컴퓨터정  
보통신공학부 학사 과정

### ▶ 박 영 철 (Young-Cheol Park)



1986년 2월: 연세대학교 전자공학과 학사  
1988년 2월: 연세대학교 전자공학과 석사  
1993년 2월: 연세대학교 전자공학과 박사  
2002년 3월 ~ 현재: 연세대학교 컴퓨터정  
보통신공학부 교수