# Convolutional Generative Adversarial Networks with Binary Neurons for Polyphonic Music Generation

**Hao-Wen Dong and Yi-Hsuan Yang**

Research Center for IT Innovation, Academia Sinica, Taipei, Taiwan

**[Demo Website]** https://salu133445.github.io/bmusegan/

## >> Introduction

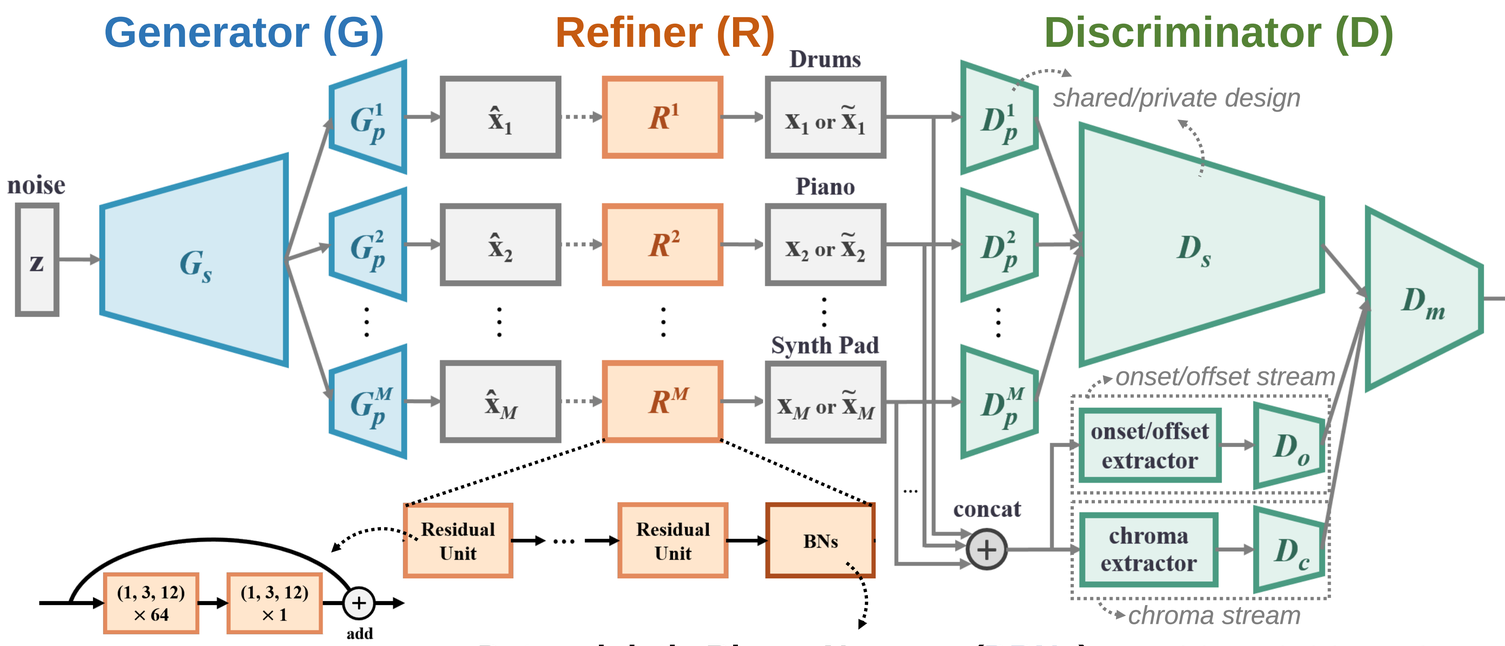
- MuseGAN [1] shows the promise of using GANs [2] with CNNs to generate ***multitrack pianorolls***. But it requires further postprocessing at test time to binarize the generator's (G) output
- **BinaryMuseGAN (proposed)** adopts ***binary neurons*** [3] to binarize G's output during training

| | G's output | data |
|---|---|---|
| **MuseGAN [1]** | real | binary |
| **BinaryMuseGAN (proposed)** | **binary** | binary |

## >> Data

- Lakh Pianoroll Dataset (LPD) — *LPD-cleansed* subset
- Consider only songs with an ***alternative*** tag to make the training data cleaner
- 13,746 **4**-bar phrases from 2,291 songs
- **96** time steps in a bar, **84** possible pitches (C1 to B7)
- **8** tracks — Drums, Piano, Guitar, Bass, Ensemble, Reed, Synth Lead and Synth Pad
- Target output tensor shape — (**4**, **96**, **84**, **8**)

## >> System




**Deterministic Binary Neurons (DBNs)** $DBN(x) = u(\sigma(x) - 0.5)$

**Stochastic Binary Neurons (SBNs)** $SBN(x) = u(\sigma(x) - v), v \sim U[0,1]$

### Training Strategies

| | |
|---|---|
| **two-stage (proposed)** | [stage 1] pretrain G and D<br>[stage 2] train R and D (G is fixed) |
| **end-to-end** | train G, R and D jointly |
| **joint** | [stage 1] pretrain G and D<br>[stage 2] train G, R and D |

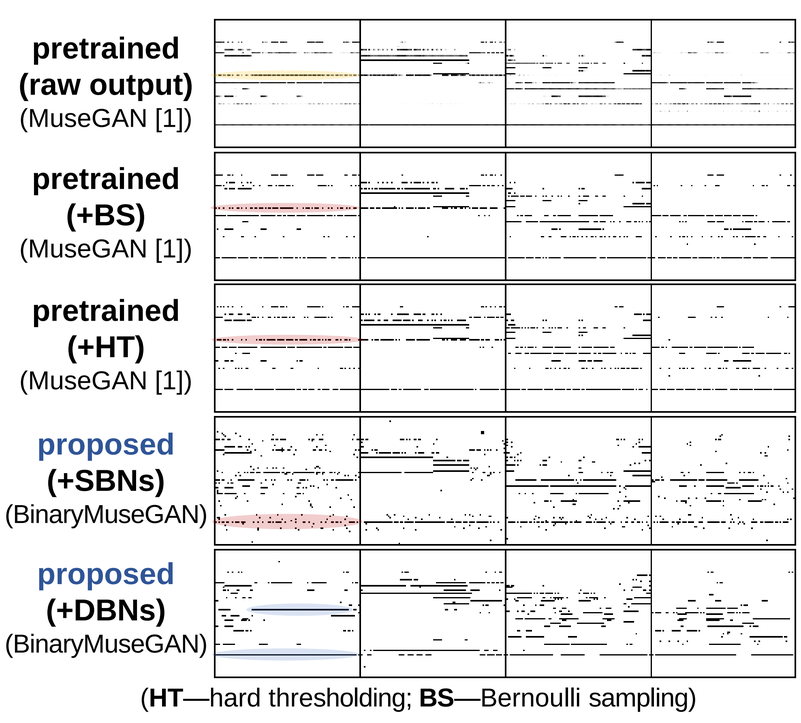
### Models

| | proposed | ablated-I | ablated-II |
|---|---|---|---|
| shared/private design | ✓ | ✓ | |
| onset/offset stream | ✓ | | |
| chroma stream | ✓ | | |

### Metrics

| | |
|---|---|
| **QN** | qualified (no shorter than a 32th note) note rate |
| **PP** | polyphonicity (more than 3 pitches are played) rate |
| **TD** | tonal distance [4] between the piano and guitar |

## >> Results



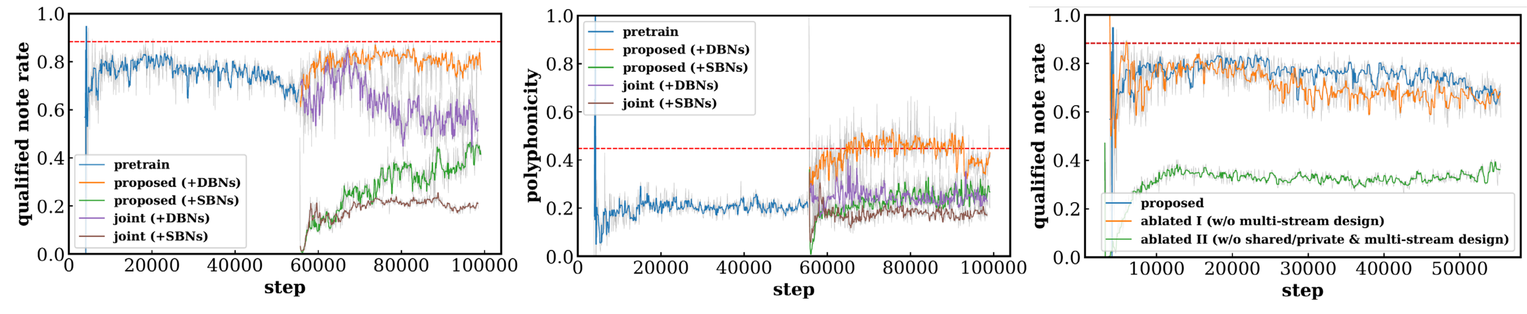

(**HT**—hard thresholding; **BS**—Bernoulli sampling)

### Evaluation Results

(value closer to the training data is better)

| | training data | pretrained | | proposed | | joint | | end-to-end | | ablated-I | | ablated-II | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | BS | HT | SBNs | DBNs | SBNs | DBNs | SBNs | DBNs | BS | HT | BS | HT |
| **QN** | 0.88 | **0.67** | **0.72** | 0.42 | **0.78** | 0.18 | 0.55 | **0.67** | 0.28 | 0.61 | 0.64 | 0.35 | 0.37 |
| **PP** | 0.48 | 0.20 | 0.22 | **0.26** | **0.45** | 0.19 | 0.19 | 0.16 | **0.29** | 0.19 | 0.20 | 0.14 | 0.14 |
| **TD** | 0.96 | **0.98** | 1.00 | **0.99** | 0.87 | **0.95** | 1.00 | 1.40 | 1.10 | 1.00 | 1.00 | 1.30 | 1.40 |

(Underlined and bold font indicate respectively the top and top-three entries with values closest to those shown in the 'training data' column.)




## >> Conclusions

- While the generated results appear preliminary and lack musicality, we showed the potential of adopting binary neurons in a music generation system
- Using DBNs leads to better objective scores than hard thresholding, Bernoulli sampling and SBNs
- It might also be interesting to use binary neurons in music transcription (binary-valued outputs as well)

## >> References

[1] Hao-Wen Dong, Wen-Yi Hsiao, Li-Chia Yang, and Yi-Hsuan Yang. MuseGAN: Symbolic-domain music generation and accompaniment with multi-track sequential generative adversarial networks. In *Proc. AAAI*, 2018.

[2] Ian J. Goodfellow et al. Generative adversarial nets. In *Proc. NIPS*, 2014.

[3] Yoshua Bengio, Nicholas Léonard, and Aaron C. Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

[4] Christopher Harte, Mark Sandler, and Martin Gasser. Detecting harmonic change in musical audio. In *Proc. ACM Workshop on Audio and Music Computing Multimedia*, 2006.