Symbolic music style transfer via latent space transformations: model and evaluation

Lucas Somacal[1*] Pablo Riera[1,2] Diego Fernández Slezak[1,2] Martín Miguel[3]

[1] University of Buenos Aires. Faculty of Natural and Exact Sciences. Computer Science Department. Buenos Aires, Argentina.
[2] CONICET-University of Buenos Aires. Computer Science Institute (ICC). Buenos Aires, Argentina.
[3] McMaster University, Ontario, Canada

[*]**Corresponding author:** lsomacal@dc.uba.ar

TL;DR

GOAL: change the style of music in symbolic format to mimic a specific music style and present new evaluation methods.

Two VAEs for symbolic musical style transfer achieve **resemblance to the target style**, **musicality**, and **identity preservation**.

3' read

Previously…

Previous **symbolic style transfer** work:

- Model transfer **from 1 specific source style to 1 specific target style** (Generative Adversarial Networks (GANs) [1], Variational Autoencoders (VAEs) [2])
- Models generate continuations for an input musical fragment in a specific style (DeepJ [3], MuseNet [4])

Previous **evaluation methods**:

- subjective listening tests
- comparison of the distributions of features to assess musicality [3]
- comparison of the predictions of style classifiers [1, 2]

What's new?

We propose:

- to do **multi-style transfer** with a **single model**,
- doing **latent space vector arithmetic**,
- to adjust the transformation level with a parameter α ∈ (0, 1],

New evaluation methods on three distinct aspects:

- whether the generated fragment presents the **target musical style**
- whether the generated fragment is **musical**
- and whether the generated fragment still **resembles the input**.

Did it work?

We evaluated two models on a specific dataset (KernScores):

- A model trained on a large dataset (Lakh) [***pre***]
- A fined tuned version of ***pre*** on the evaluation dataset [***fine***]

Both models managed to produce new music that was **closer to the target style**, was **musical** and **preserved the identity** of the original music fragment.

The ***fine***-tuned model **performed slightly better** than **pre**.

Methods

Datasets

- **Lakh Midi Dataset** [5]: classic pop and rock, pop, folk or **classical** (tags from musicbrainz.org) – 155,037 music fragments.
- **KernScores** [6] (fine-tuning and evaluation): *Bach's* chorals, *Frescobaldi's* canzoni, *Mozart* piano sonatas and *ragtimes* – 2032 fragments.
- **Validation set**: 10% of KernScores.

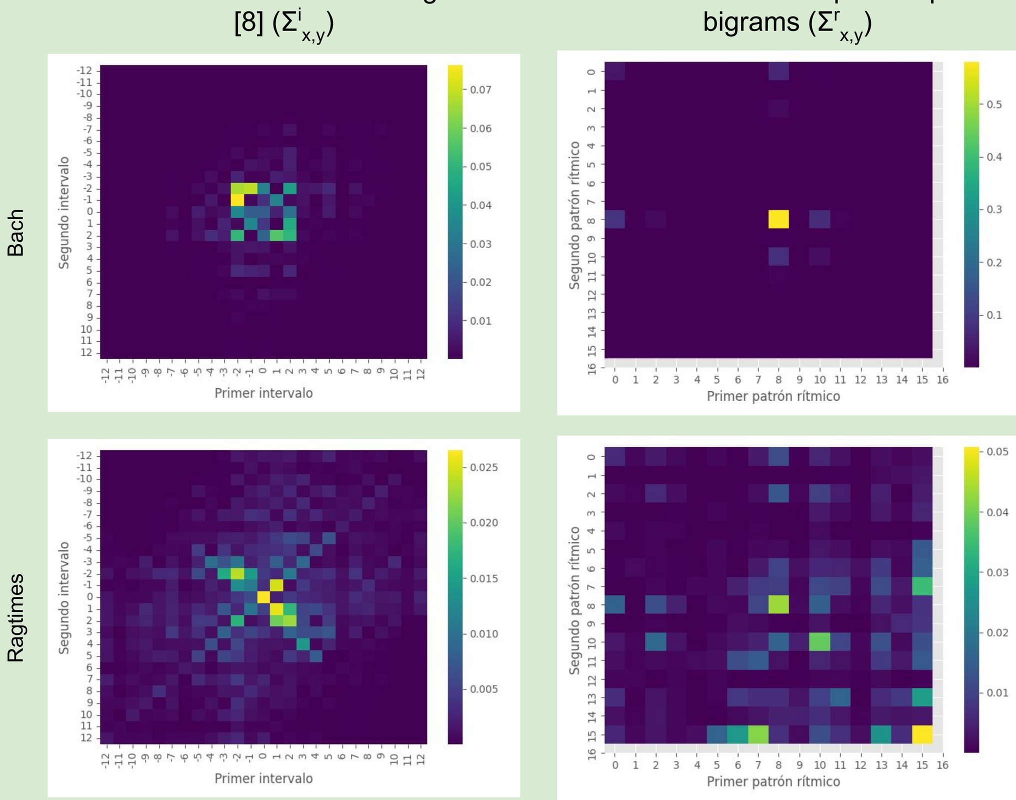
Music representation and model

- **Input**: matrices of 1s and 0s with **64 rows as time units** (semiquavers, spanning 4 bars) and **89 columns indicating pitch and note changes** (rhythm).
- Adaptation of the model from [7] (a VAE):
  - Encoder: 2 **bi-GRU** + 2 **Dense**
  - **Latent space**: **96** dimensions
  - Decoder: **Repeat vector** + 2 **GRU** + **Dense**.
- We trained two models:
  - ***Pre*** fine-tuning: based on Lakh Midi Dataset.
  - ***Fine***-tuning: fine-tuned in KernScores.

Modeling of musical style

Melody — Distribution of note-interval bigrams [8] ($\Sigma^i_{x,y}$)

Rhythm — Distribution of 4 semi-quavers pattern bigrams ($\Sigma^r_{x,y}$)



Style transfer process

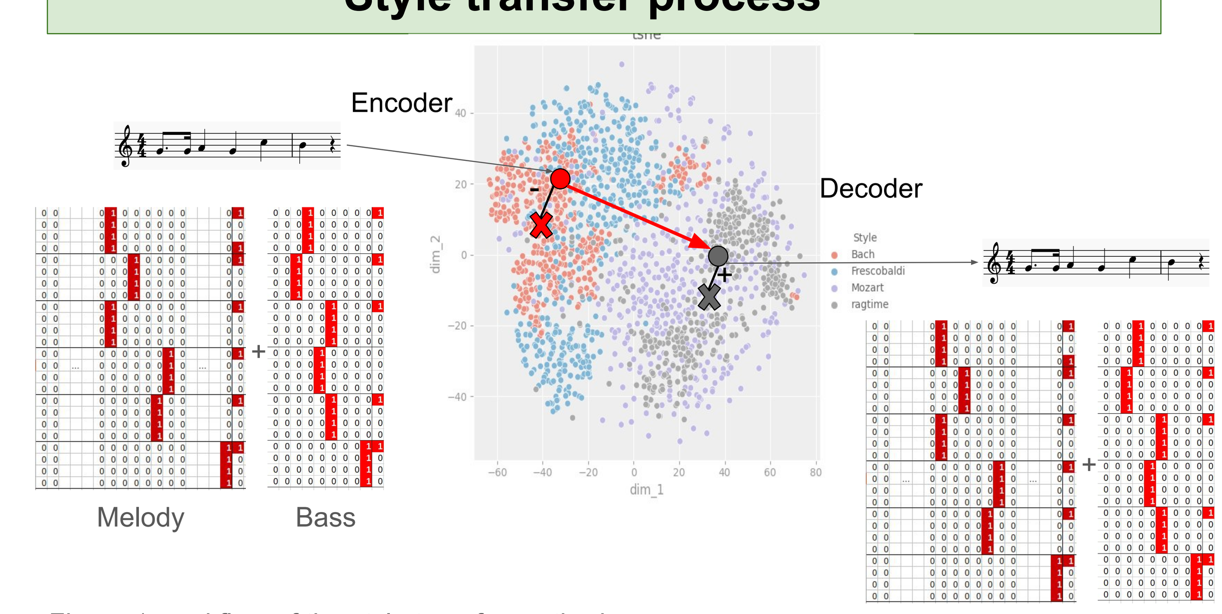


Figure 1: workflow of the style transfer method.

1. We **encode** 64x89 binary matrices representing two tracks (melody and bass) of a fragment of 64 semiquavers of music.
2. We **add** the characteristic vector $v_{s'}$ of the **target style** and **subtract** the characteristic vector v_s of the **original style** weighted by **α ∈ (0,1]**.
3. We **decode** it to obtain the new fragment.

$$t_{s,s'}(m) = decode(encode(m) + \alpha(v_{s'} - v_s))$$

Evaluation and results

1. The generated fragment belongs to the target style?

A transformation is **successful** if the generated fragment m' is **closer to the target style** *s* **than the original fragment** m, that is:

$$\Delta(m', M_s) < \Delta(m, M_s)$$

- We measure the **distance** between a fragment and a style with ***optimal transport***
- For each pair of source-target styles we calculate the percentage of generated fragments that became closer to the target style.

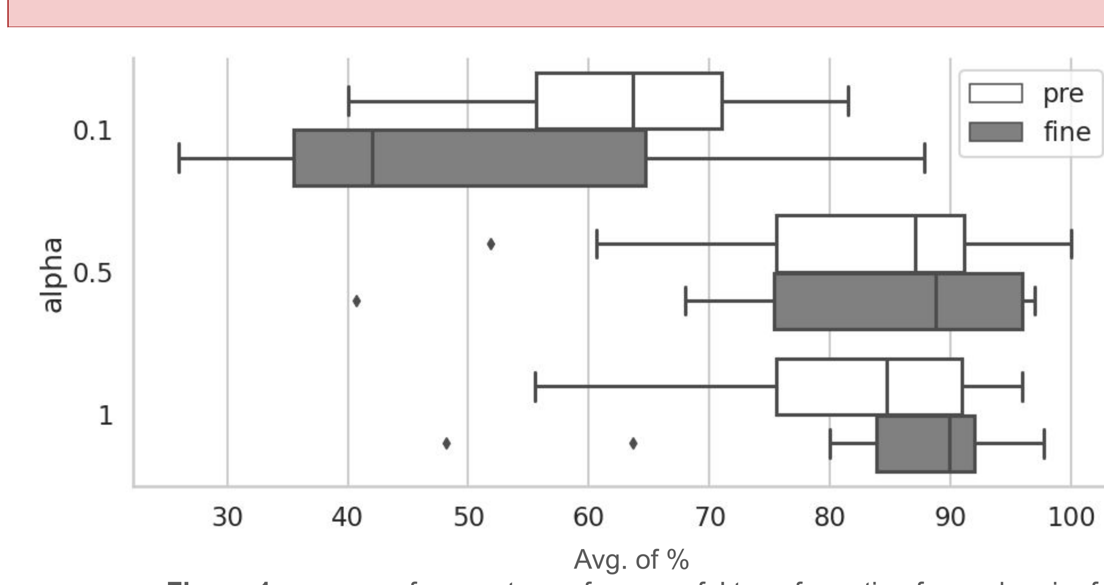


Figure 1: average of percentage of successful transformation for each pair of styles (for each alpha value and models)

- As ***α* gets larger**, more transformed **fragments are closer** to the target style.
- There are no noticeable differences between the *pre* and *fine* models, except with the small *α*, where *pre* performs better.

2. The generated fragment remains musical?

Musicality: the percentage of permutations that are less likely sampled (δ) from an **universal style**.

- We consider a ***universal style*** M_u formed by the balanced sum of the fragments of the different styles of a dataset:
- For each original fragment we generated 20 permutations by reordering the notes in time.
- *The sampling likelihood is defined as:*

$$\delta(m, M_u) = \Sigma_{x,y} \log(\Sigma^i_{x,y}(M_u))\, \sigma^i_{x,y}(m) + \Sigma_{x,y} \log(\Sigma^r_{x,y}(M_u))\, \sigma^r_{x,y}(m)$$

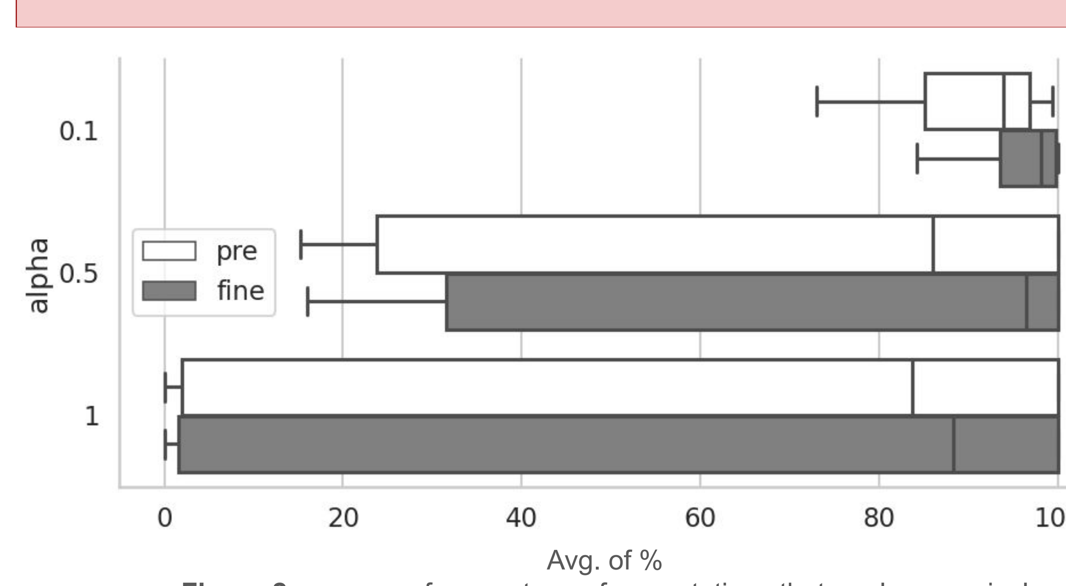


Figure 2: average of percentage of permutations that are less musical than *m'* for each style pair of styles (for each alpha value and models).

- A **larger α** yields a larger transformation, which may yield **less musical** results (Nonetheless, most cases are above 80%).
- *Fine-tuned* model performs slightly better than *pre.*

3. The generated fragment is similar to the original?

m' **retained characteristics** of *m,* the higher it appears in the **similarity ranking**.

- We propose a **similarity ranking** between *m* against the set composed of *m'* and all other fragments of the original style.
- Two fragments' similarity is the inverse of **how many semitones the notes differ** between one fragment and the other for each time instant (a rest compared with a note is considered as 12).
- The score is bound by 0 and 1 where 1 is the best value.

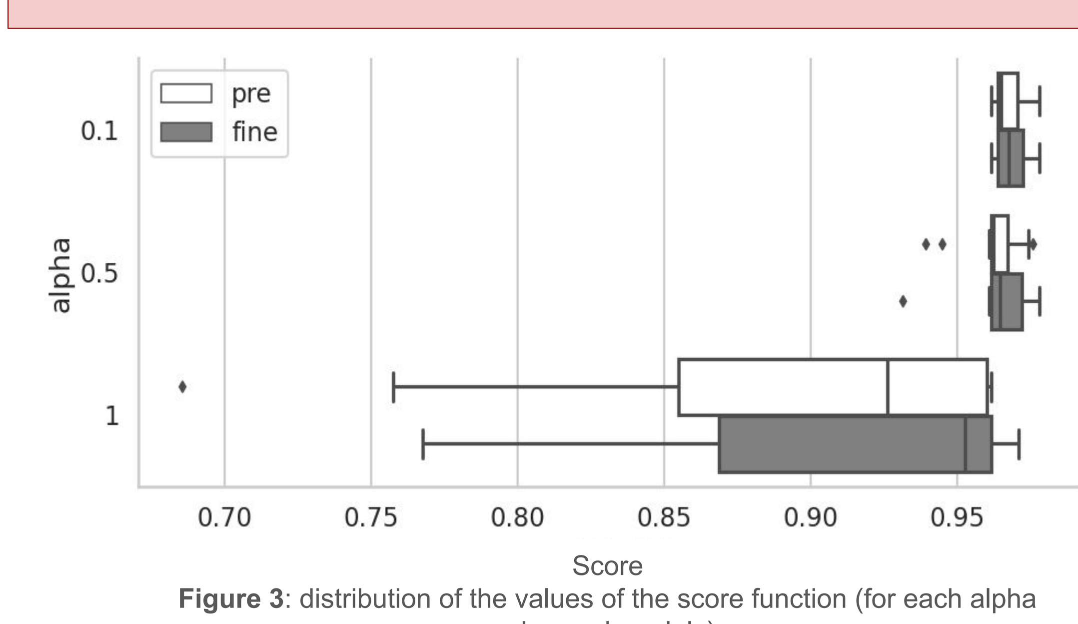


Figure 3: distribution of the values of the score function (for each alpha value and models).

- With a **large α** value the **performance is worse** but still good.
- *Fine-tuned* model performs slightly better than *pre.*

Conclusions

- Our models managed to generate new fragments that **remained musical**, kept the **identity of the original fragment** and that were also **closer to the target style**.
- This happened for **certain values of *α*** (0.1 and 0.5).
- A greater *α* implies more style approach but less musicality.
- The model trained on a general music dataset was successful even on the distinct set of evaluation styles.
- When observing the **performance between specific source-target style pairs**, we noticed performance **varied**.
- The model struggled to transform between Mozart and Ragtime, contrary to our expectation that styles with similar complexity would yield better results.
- As **future work**, we suggest **validating** our proposed metrics with **listener surveys** and **compare** our metrics with those used in **previous work**.
- Our transformation method could benefit from the **latent space disentanglement** to represent style.
- Compare the style-specific vs. general approaches.

References

[1] G. Brunner, Y. Wang, R. Wattenhofer, and S. Zhao, "Symbolic music genre transfer with cyclegan", 2018.
[2] G. Brunner, A. Konrad, Y. Wang, and R. Wattenhofer, "MIDI-VAE: modeling dynamics and instrumentation of music with applications to style transfer", 2018.
[3] H. H. Mao, T. Shin, and G. Cottrell, "Deepj: Style-specific music generation", 2018.
[4] C. Payne, "Musenet," openai.com/blog/musenet, 2019.
[5] C. Raffel, "Learning-based methods for comparing sequences, with applications to audio-to-midi alignment and matching", 2016.
[6] "Kernscores," https://kern.humdrum.org/.
[7] R. Guo, I. Simpson, T. Magnusson, C. Kiefer, and D. Herremans, "A variational autoencoder for music generation controlled by tonal tension", 2020.
[8] P. Zivic, F. Shifres, and G. Cecchi, "Perceptual basis of evolving western musical styles", 2013.