

Génération de musique et de transitions via auto-encodeur variationnel

Geoffrey Saunois, Victor Saillant, Matheus Coutiers, Benoit Maïzi

09 / 03 / 2021

Plan



1. Objectifs du projet
2. Solution mise en oeuvre
3. Implémentation du modèle
4. Transition de musique
5. Résultats

A. Annexes

1. Objectifs du projet

Entraîner un modèle d'apprentissage profond sur des mélodies simples, dans l'objectif de :

1. Générer de nouvelles mélodies
2. Étant donné deux mélodies, être capable de générer une transition entre les deux (mélodie 1 \rightarrow mélodie 2)





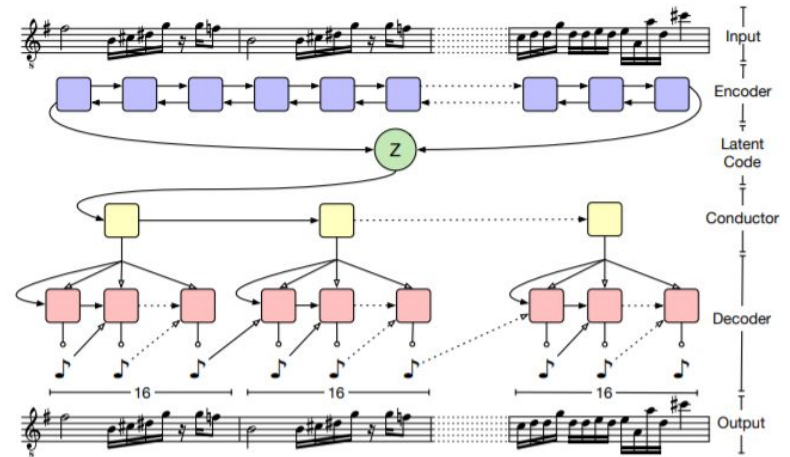
2. Solution mise en oeuvre - *Recherche de modèle*

- Recherche d'un modèle génératif → Auto-encodeur variationnel
- Traitement de séquences → Utilisation de LSTM dans l'architecture

2. Solution mise en oeuvre - *Modèle*

Auto-encodeur variationnel, basé sur [1] (A Hierarchical Latent Vector Model for Learning Long-Term Structure in Music).

```
VariationalAutoencoder(  
    (encoder): LSTM(61, 256, batch_first=True, bidirectional=True)  
    (encoderOut): Linear(in_features=512, out_features=128, bias=True)  
    (linear_z): Linear(in_features=64, out_features=32, bias=True)  
    (dropout): Dropout(p=0.2, inplace=False)  
    (worddropout): Dropout2d(p=0.2, inplace=False)  
    (conductor): LSTM(32, 32, batch_first=True)  
    (decoder): LSTM(93, 32, batch_first=True)  
    (linear): Linear(in_features=32, out_features=61, bias=True)  
)
```



3. Implémentation du modèle - *Données*

Quelles données ?

MP3, WAV, MIDI
Utilisation pratique

pretty_midi

Lahk MIDI Dataset,
autre dataset

Entrée du VAE

Preprocessing

quelle(s) piste(s) retenir ?

traitement des pistes :

trim_blank, chopster/chop,
arpster/arpeggio, cutster/measure maker,
padster/padding, transposer (mineur et
majeur)

formatage des échantillons

Limitations

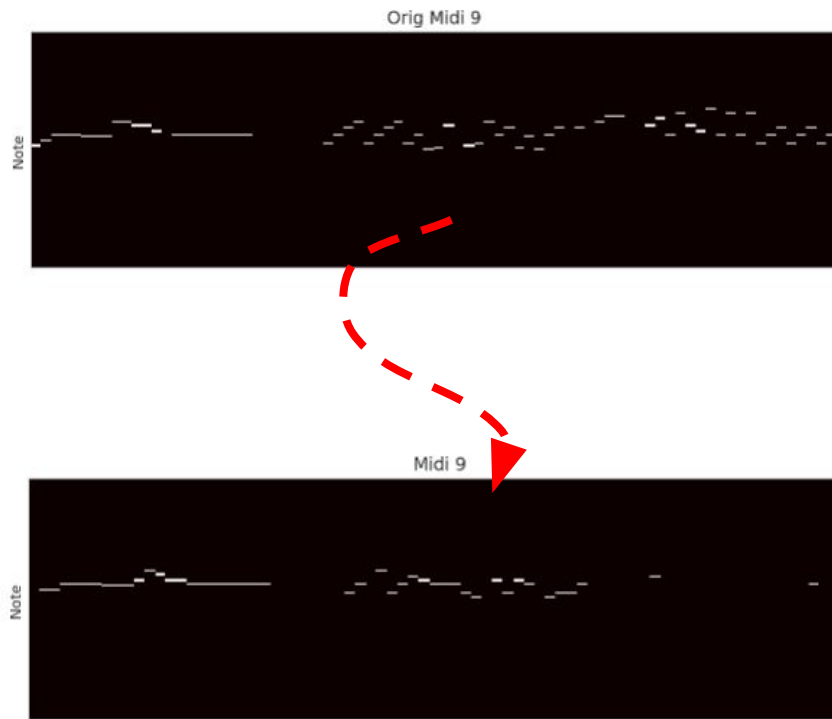
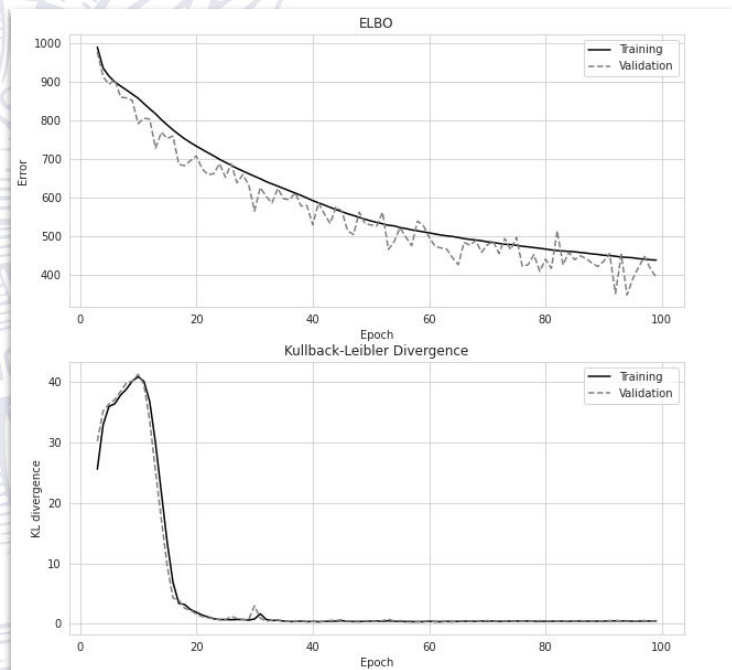
Temps d'écriture des
fichiers MIDI vers CSV

Qualité des données
(piste de “piano”)

Point de départ :

<https://github.com/Variational-Autoencoder/MusicVAE>

3. Implémentation du modèle - *Résultats en reconstruction*



4. Transition de musique - 3 *stratégies*

Stratégie 1:

Reconstruction à partir d'une combinaison linéaire des embeddings des deux mélodies

$$z[t] \leftarrow t * z_1[t] + (1 - t) * z_2[t]$$

Stratégie 2:

Embedding de la première mélodie

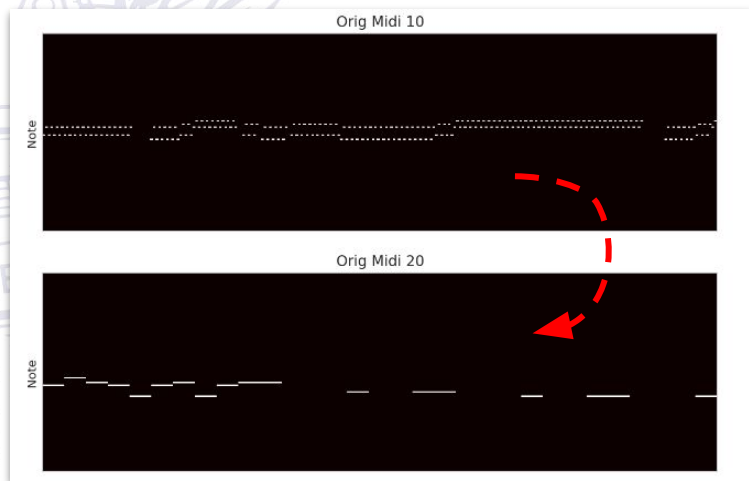
Teacher Forcing avec une combinaison probabiliste des deux mélodies

$$\begin{aligned}\mathbb{P}(m[t] = m_1[t]) &\sim \mathcal{B}(1 - t) \\ \mathbb{P}(m[t] = m_2[t]) &\sim \mathcal{B}(t)\end{aligned}$$

Stratégie 3:

Stratégie 1 + 2

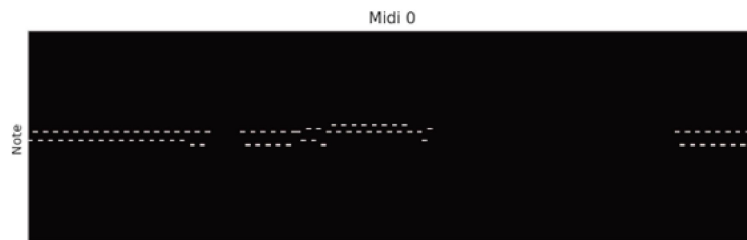
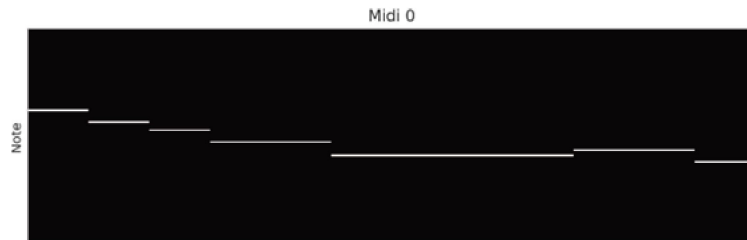
4. Transition de musique - *Résultats*



stratégie 1

stratégie 2

stratégie 3



Perspectives

Améliorer la transition

Echelle du mélange des
données de contrôle avec
teacher forcing pour la
transition :

mesure, i.e. 16 notes



note par note

Utiliser deux
conductors LSTM
associés à chaque
mélodie pour réaliser
la transition

Autre sujet intéressant

la complétion, la
poursuite d'une mélodie

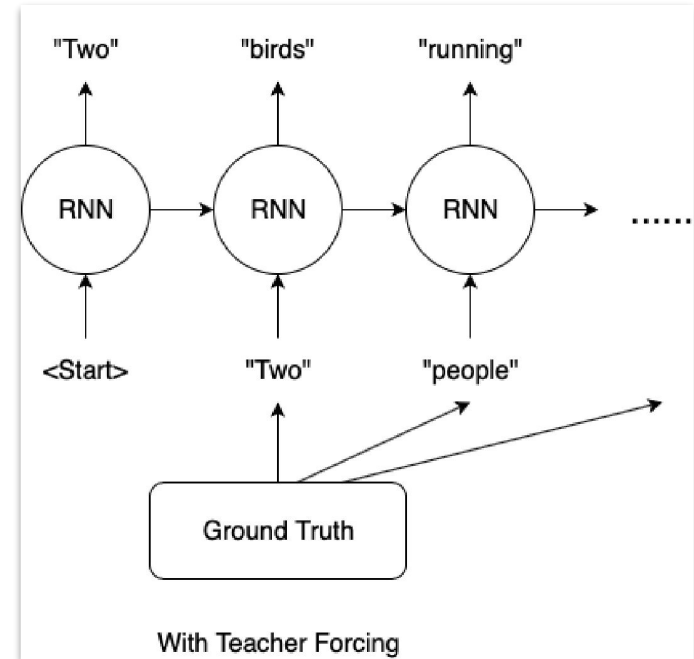
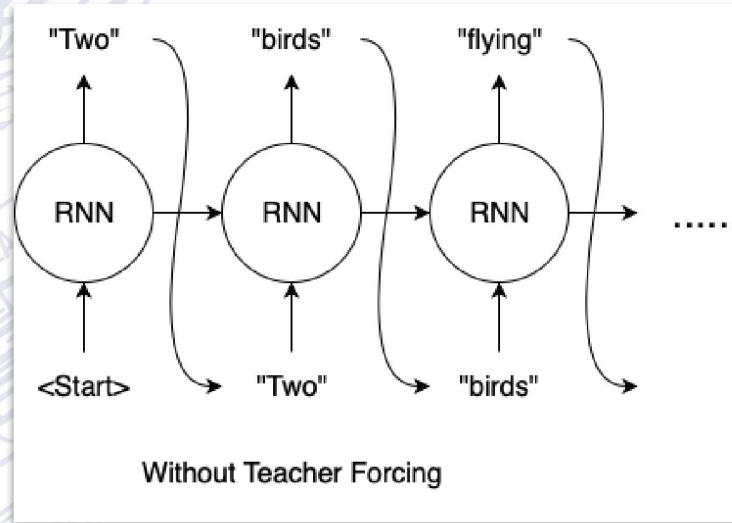


Merci de votre attention

<https://github.com/bison-fute/music-VAE>

Matheus Coutsiers, Benoit Maizi, Victor Saillant, Geoffrey Saunois

A. Annexes - *qu'est ce que le teacher forcing ?*



→ Même principe avec des LSTM

A. Annexes - *génération de musique : le prochain Mozart ?*

