

Rapport (sprint 2) du projet de 8INF892

Wenhao LUO
LUOW09129805

April 7, 2023

1 Introduction

Dans le cadre du cours 8INF892, on a besoin de réaliser un projet en utilisant les outils de l'apprentissage profond. On choisit de créer un réseau neurone qui permet de produire de la musique en imitant le style de Bach. Le projet sera réaliser en utilisant `Python`, avec l'aide de la bibliothèque `TensorFlow`.

Pour le deuxième sprint, on a progressé en effectuant plusieurs tâches. Un bilan des tâches réalisées est présenté dans le tableau 1. Dans les prochaines sections, on va présenter en détails des tâches réalisées.

Le dépôt du projet est disponible sur Github :

<https://github.com/LuoQuestionmark/projet-appentissage-profond>.

nom de tâche	état
préparation de données (premier modèle)	terminé
création du réseau (premier modèle)	terminé
préparation de données (deuxième modèle)	en train de faire
création du réseau (deuxième modèle)	à faire
mélange des résultats	à faire
présentation des résultats	à faire

Table 1: Le bilan sur les tâches réalisées ainsi que le plan pour le suivant.

2 Premier modèle

2.1 Préparation des données

Dans le rapport du sprint I, nous avons mentionné l'utilisation de musiques de Bach au format `midi` comme données d'entraînement. À présent, nous avons divisé ces tâches en plusieurs étapes. Un premier programme a été développé pour lire et analyser les fichiers ; un deuxième programme permet de traduire ces données sous les format "one hot", les données préparées sont ensuite utilisées pour l'entraînement du modèle.

L'ensemble des notes est enregistré par leur temps de début t_s , leur temps de fin t_e , et leur hauteur tonale p . On peut définir un espace $\mathbb{E} = \mathbb{N} \times \mathbb{N}$, une injection f , et toutes les notes peuvent être projetées cet espace par leur temps de début et leur hauteur tonale. On peut donc définir la notion de distance sur cet espace \mathbb{E} . La distance entre deux notes n_1, n_2 est définie par $d = \|n_1 - n_2\|_2$. Cette notion permet de déduire des "voisins" d'une note donnée.

2.2 Entraînement du modèle

Si une note n_1 possède une liste de voisins $(n_i)_{i \in [2, n]}$, l'ensemble des notes peut être considéré comme une entrée du modèle. La liste des voisins est la variable, et la note n_1 est le label. Le but de ce premier modèle est de prédire une note

manquante en utilisant tous ses voisins. À partir de ce modèle, le système peut imiter le style du compositeur.

3 Deuxième modèle

Le problème avec le premier modèle est le manque de connaissances spatiales. Actuellement, nous sommes en train de préparer un deuxième modèle qui permet d’encoder ces connaissances. Nous avons trouvé un article qui utilise des réseaux neuronaux RNN et LSTM pour effectuer une recherche similaire[1].

4 Présentation de la sortie des réseaux

En utilisant le premier modèle, nous sommes déjà capables de créer des notes musicales à partir de quelques notes données. Pour l’instant la sortie n’est pas vraiment lisible, un des exemples est donné dans le listing 1. Chaque fois le système essaie de prédire une note, et le résultat est marqué comme “drop”.

```
1 1/1 [=====] - 1s 1s/step
2 drop: 55
3 1/1 [=====] - 0s 15ms/step
4 drop: 60
5 1/1 [=====] - 0s 17ms/step
6 drop: 62
7 1/1 [=====] - 0s 16ms/step
8 drop: 64
9 1/1 [=====] - 0s 14ms/step
10 drop: 59
11 1/1 [=====] - 0s 14ms/step
12 drop: 67
13 1/1 [=====] - 0s 14ms/step
14 drop: 55
15 1/1 [=====] - 0s 16ms/step
16 drop: 62
17 1/1 [=====] - 0s 15ms/step
18 drop: 64
19 1/1 [=====] - 0s 17ms/step
20 drop: 65
21 1/1 [=====] - 0s 15ms/step
```

Listing 1: Une sortie du premier modèle.

Ce n’est donc pas un format lisible pour humain ; pour le rendu final du projet, on planifie à utiliser le format du Lilypond comme la sortie. C’est un logiciel du projet GNU¹.

¹<http://lilypond.org/>

5 Conclusion

Dans le deuxième sprint, on a réalisé un premier modèle et on prévoit d'utiliser un deuxième modèle pour la suite. On a aussi traité des données et on a retrouvé un format pour la sortie de notre programme.

References

- [1] Zheng Sun, Jiaqi Liu, Zewang Zhang, Jingwen Chen, Zhao Huo, Ching Hua Lee, and Xiao Zhang. Composing music with grammar augmented neural networks and note-level encoding. In *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1864–1867, 2018.