靜 宜 大 學

資訊工程學系

畢 業 專 題 成 果 報 告 書

專題名稱

ＡＩ音樂產生器

學 生：

資工四 A　411030630　尤睿杰

資工四 B　411017836　文睿薪

指導教授：林耀鈴　教授

西 元 二 **O** 二四 年 十二 月

×××××××××××××××（ＡＩ音樂產生器）×××××××××××××××

學生：尤睿杰　　指導教授：林耀鈴
文睿薪

靜宜大學資訊工程學系

摘　　要

本專題旨在設計並實現一個基於人工智慧的音樂生成系統，利用 Transformer 架構的深度學習技術，探討如何生成具創意性與結構化的音樂旋律。本系統主要依據 Music Transformer（Huang et al., 2018）的模型設計，採用了 PyTorch 平台進行模型實現與優化，並引入相對位置表示（Relative Position Representation, RPR）以提升長序列依賴的學習能力。

研究過程包括資料收集、預處理、模型訓練與生成評估。資料來源為 Maestro V2 數據集，其包含大量高品質的 MIDI 格式音樂檔案，涵蓋多種音樂類型與風格。首先，我們對數據進行標準化處理，包括去除持續踏板（sustain pedal）的錯誤訊號、分割數據為訓練、驗證與測試集，並將其轉化為模型可讀的序列格式。接著，我們利用優化過的 Music Transformer 模型進行訓練，模型結構包含 6 層編碼器，8 頭多頭注意力機制，每層的模型維度為 512，前饋神經網絡的維度為 1024，並加入了 0.1 的隨機丟棄率（dropout）以防止過擬合。

在訓練過程中，我們使用了相對位置表示（RPR）改進模型的性能，使其更適應長序列的音樂生成任務。模型經過 100 個訓練周期，損失值（Loss）從初始的 6.0 快速下降到穩定於 3.995，顯示出良好的收斂性。生成過程中，我們採用隨機取樣策略（beam = 0），基於提供的初始旋律片段（num_prime =

256）生成長度為 1024 的 MIDI 音樂序列。

實驗結果顯示，生成的音樂旋律在和諧性、連貫性與創意性方面均表現良好，可作為遊戲配樂、教育工具或其他娛樂應用的潛在方案。模型的改進方法（RPR）有效提升了音樂結構的表現力，並減少了重複音符的出現。未來研究將著重於數據增強技術的應用、其他數據集的整合，以及固定長度音樂生成策略的改進，以進一步提升系統的實用性與生成效果。

靜宜大學資訊工程學系
專題實作授權同意書

本人具有著作財產權之論文全文資料，授予靜宜大學資工系，為學術研究之目的以各種方法重製，或為上述目的再授權他人以各種方法重製，不限地域與時間，惟每人以一份為限。授權內容均無須訂立讓與及授權契約書。依本授權之發行權為非專屬性發行權利。依本授權所為之收錄、重製、發行及學術研發利用均為無償。

指導教授 _____林耀鈴_____

學生簽名:尤睿杰	學號:411030630	日期:西元	年	月	日
學生簽名:文睿薪	學號:411017836	日期:西元	年	月	日

指導教師簽章 _____

西 元 **2024** 年 **12** 月 日

靜宜大學資訊工程學系

專題實作指導教師確認書

茲確認專題書面報告之格式及內容符合本系之規範

畢業專題實作名稱：_____AI 音樂產生器_____

畢業專題實作分組名單：　共計 __2__ 人

組員姓名	學號
尤睿杰	411030630
文睿薪	411017836

指導教師簽章 _____

西 元 **2024** 年 **12** 月 日

誌　　謝

_____

_____

_____

_____

_____

目　　錄

`

一、緒論

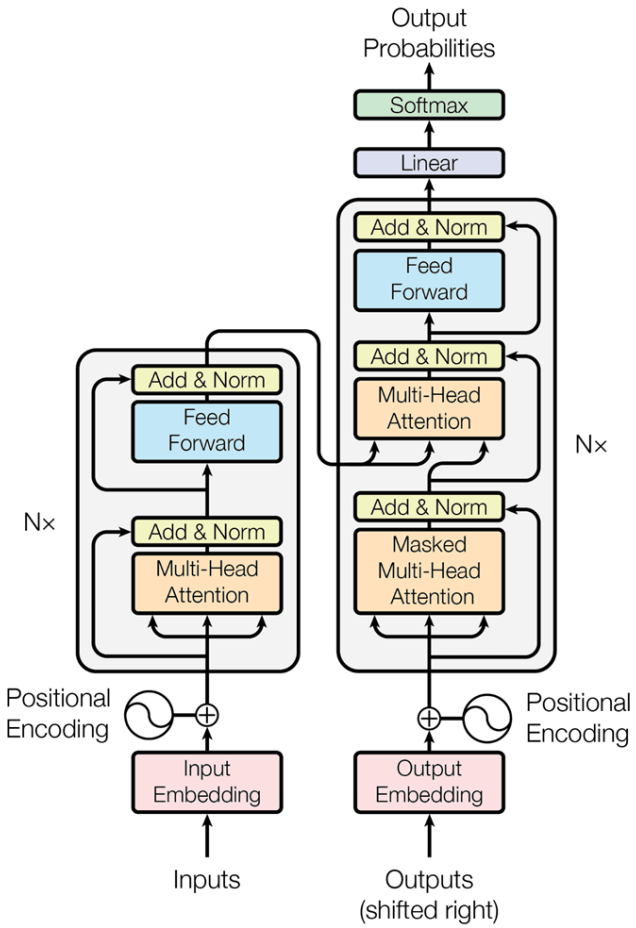
1.1 研究背景與動機

音樂生成技術是人工智慧領域的一個重要研究方向，其應用範圍包括娛樂、教育、創意產業等多個領域。隨著深度學習技術的快速發展，基於 Transformer 架構的生成模型在自然語言處理領域取得了卓越成果，這一技術也逐漸被應用於音樂生成領域。Music Transformer 模型（Huang et al., 2018）是其中的代表作，其特點在於能有效捕捉長序列依賴，生成具有結構化特徵的音樂旋律。然而，在生成過程中，如何進一步提升生成音樂的和諧性與多樣性，仍然是一項挑戰。本專題旨在針對這些挑戰進行深入研究與改進，並設計一個高效、實用的音樂生成系統。

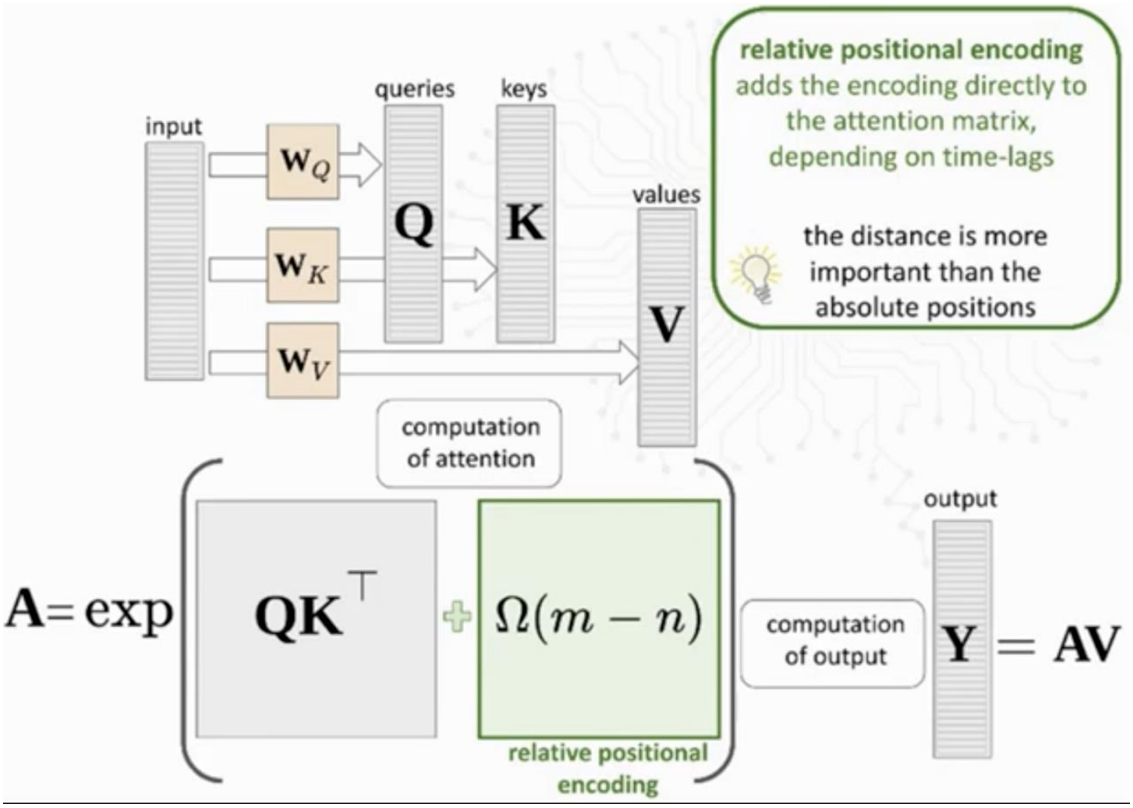
1.2 研究目的與目標

本專題的目的是設計並實現一個基於人工智慧的音樂生成系統，利用 Transformer 架構如(圖一)及相對位置表示（RPR）技術如圖(二)，探索如何生成具創意性與結構化的音樂旋律。目標包括：

- 開發一個能生成長序列 MIDI 音樂的系統。
- 提升生成音樂的和諧性、連貫性與多樣性。
- 將生成系統應用於遊戲配樂、教育工具等實際場景。


Output Probabilities
Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
N×
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
N×
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)

(圖一)


relative positional encoding
adds the encoding directly to the attention matrix, depending on time-lags
the distance is more important than the absolute positions
input
queries
keys
values
$\mathbf{W}_Q$
$\mathbf{W}_K$
$\mathbf{W}_V$
$\mathbf{Q}$
$\mathbf{K}$
$\mathbf{V}$
computation of attention
$\mathbf{A} = \exp\left(\mathbf{Q}\mathbf{K}^\top + \Omega(m-n)\right)$
relative positional encoding
computation of output
output
$\mathbf{Y} = \mathbf{A}\mathbf{V}$

圖(二)

1.3 研究範圍與限制

本專題研究範圍包括基於 MIDI 格式音樂數據的生成技術，重點聚焦於採用 Music Transformer 模型進行長序列音樂生成。研究的限制包括：

- 使用的數據集僅限於 Maestro V2，可能無法涵蓋所有音樂風格。
- 模型的生成效率受到硬件資源的限制，尤其是在長序列生成任務中。
- 現階段未涉及多模態（如音頻、歌詞等）的生成應用。

1.4 報告架構概述

本報告共分為四個章節：

- 第一章 緒論：介紹研究背景、目的、範圍與限制。
- 第二章 專題內容與進行方法：描述系統設計與實施細節。
- 第三章 專題成果介紹：分析生成結果與性能。
- 第四章 專題學習歷程：總結專題進行過程中的學習經驗。
- 第五章 結論與未來展望：總結研究成果並探討未來改進方向。

二、專題內容與進行方法

2.1 研究動機與目的

音樂生成技術的研究不僅滿足了娛樂和教育的需求，也為創意產業帶來了新的可能性。隨著深度學習的快速發展，基於 Transformer 架構的模型在長序列數據建模中取得了優異的成果。然而，在音樂生成領域中，如何提升旋律的和諧性與多樣性仍是一個挑戰。本專題旨在利用相對位置表示（Relative Position Representation, RPR）技術，對 Music Transformer 模型進行改進，以解決長序列依賴的問題，並設計一個高效的音樂生成系統，從而實現具創意性與結構化的音樂生成。

2.2 技術與架構選擇

本專題採用了 Music Transformer 作為核心生成模型，該模型利用 Transformer 架構中的編碼器對音樂序列進行建模。為了改善長序列依賴的學習能力，我們引入了相對位置表示（RPR），使模型能夠更有效地捕捉音符間的相對關係。

技術選擇包括：

- **框架與工具**：使用 PyTorch 進行模型實現與訓練，並採用 Python 開發環境。
- **數據集**：採用 Maestro V2 數據集，包含多種風格的高品質 MIDI 音樂文件。
- **模型結構**：如(表一)
- **VOCAB_SIZE：390** - 決定了嵌入層和輸出層的大小，涵蓋 MIDI 序列中的所有可能鍵（token）。
- **d_model：512** - 每個鍵被嵌入到 512 維的向量空間中，捕捉鍵之間的語義關係。
- **預測過程** - 模型輸入當前序列，生成下一個鍵的概率分布（VOCAB_SIZE 維）。 - 選擇最大概率的鍵作為預測結果（單選一分類問題）。如圖(三)

重點公式: y = Softmax(W_out * H) next_token = argmax(y)

參數名稱	值
批量大小（batch_size）	2
最大序列長度（max_sequence）	2048
Transformer 層數（n_layers）	6
多頭注意力機制（num_heads）	8
模型維度（d_model）	512
前向傳遞維度（dim_feedforward）	1024

丟棄率（dropout）	0.1

（表一）

2.3 訓練與測試流程

訓練與測試流程主要分為三個階段：

1. **數據預處理**：
- 將 Maestro V2 數據集中的 MIDI 文件轉換為模型可讀的序列格式。
- 修正持續踏板（sustain pedal）錯誤，確保數據質量。

2. **模型訓練**：
- 設置訓練參數，如批量大小（batch_size = 2）、最大序列長度（max_sequence = 2048）。
- 使用 Adam 優化器進行梯度更新，並設置學習率調整策略。
- 訓練過程中觀察損失值與模型生成性能。

3. **生成與測試**：
- 使用 `generate.py` 腳本生成長度為 1024 的 MIDI 音樂片段。
- 通過主觀聆聽與客觀指標（如損失值）評估生成音樂的和諧性與創意性。

圖(三)

三、專題成果介紹

3.1 音樂生成樣本與效果分析

在完成模型訓練後，我們基於最佳權重檔案生成了多段 MIDI 音樂樣本。生成過程中採用相對位置表示（RPR）技術，生成結果在旋律結構、和諧性及多樣性方面展現了良好的表現。

此外，透過鋼琴滾動條動畫（Piano Roll Visualization），我們將生成樣本的 MIDI 文件進行可視化，便於觀察旋律的節奏分布與結構完整性。分析結果表明，模型能有效捕捉長序列音樂的特徵，並生成具創意的音樂旋律。

生成樣本分析與比較

以下是基於不同參數 `num_prime` 的生成音樂樣本描述及分析：

樣本一：

參數設置：`num_prime = 1`

生成描述：此樣本僅使用 1 個引子音符作為模型的生成起點。結果顯示，模型在極少的上下文條件下，生成了具有較高隨機性的音樂，但在結構和旋律的連貫性方面略有欠缺。

特性：創造性高，但缺乏一致性，適合測試模型的音樂想像能力。

樣本二：

參數設置：`num_prime = 256`

生成描述：此樣本使用 256 個 MIDI 消息作為生成的引子。生成結果顯示，模型能有效地捕捉引子中的旋律特徵，並在此基礎上延續音樂結構，生成的音樂具有高度連貫性和完整性。

特性：保留了引子的風格特徵，適合用於生成與原始音樂一致性高的延續作品。

樣本三：

參數設置：`num_prime = 256`

生成描述：與樣本二相似，此樣本生成的音樂具有連貫性，且更強調完整的結構。

特性：延續了引子的風格，適合生成完整的音樂片段。

如(表二)

結論：

通過對三個生成樣本的比較，我們可以看出 `num_prime` 值的不同對生成音樂的特性具有重要影響。小的 `num_prime` 值適合創造性生成，而大的 `num_prime` 值能保證風格一致性和連貫性。

樣本名稱	num_prime	生成長度	生成特性
樣本一	1	1024	隨機性強，創造性高，但結構不連貫
樣本二	256	2048	音樂連貫性強，保留了引子的風格
樣本三	256	2048	與樣本二相似，生成的完整性和結構更明顯

(表二)

生成的音樂網址

https://drive.google.com/drive/folders/1soXJ8CUqTW24z7R9ioSpM81F9cu3OerL?usp=sharing

3.2 預測過程與 VOCAB_SIZE 的應用

本專題的 Music Transformer 模型設計為單選一分類問題：

1.輸出空間：模型的輸出是一個 VOCAB_SIZE = 390 的概率分布，涵蓋所有 MIDI 音樂事件。 2.預測方式：通過 Softmax 計算概率分布，使用 argmax 選擇概率最高的鍵作為下一個輸出。

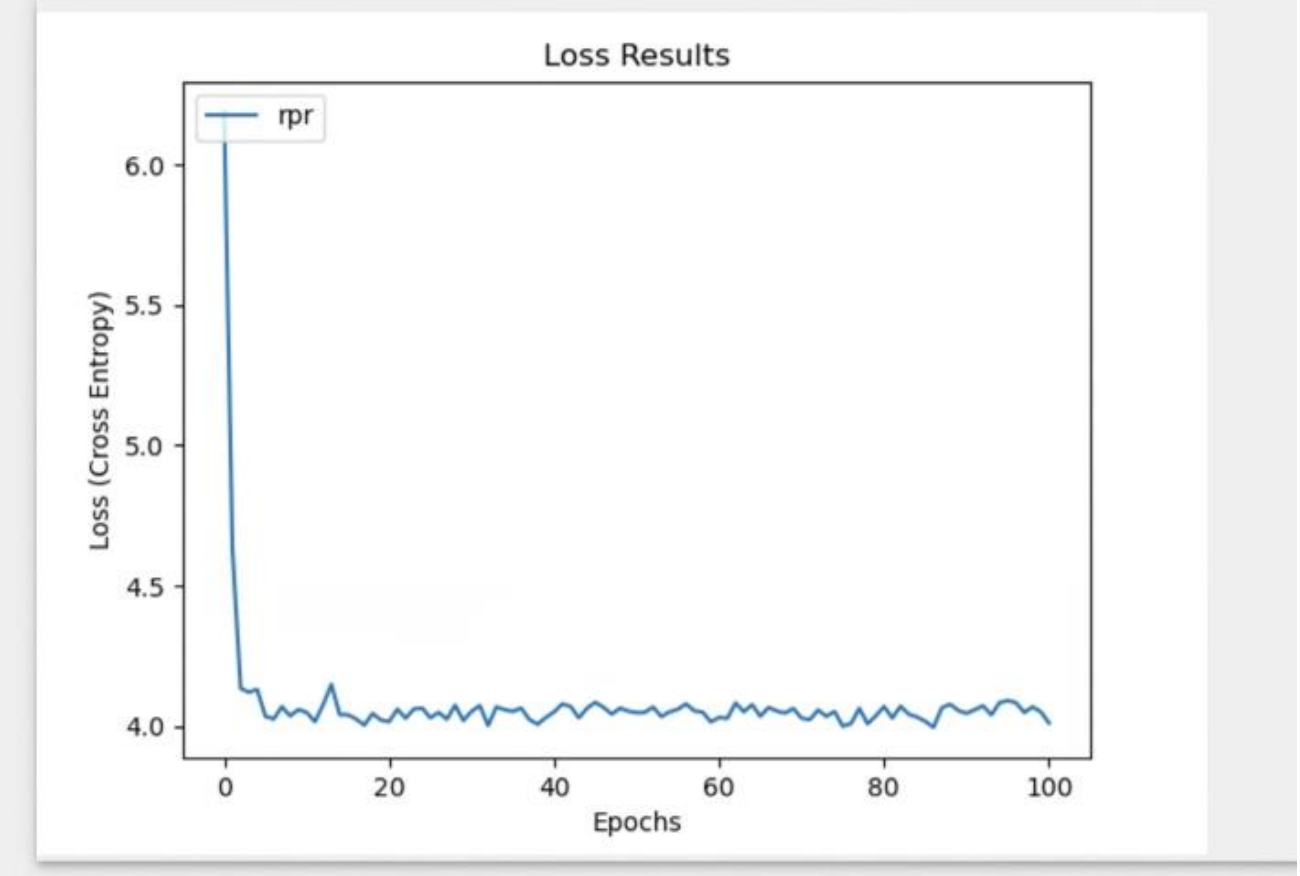
3.損失函數應用：使用 CrossEntropyLoss，優化模型對正確鍵的預測概率。

3.3 性能與準確度比較

損失值結果分析

我們使用 Music Transformer 模型對數據進行訓練並測試，最佳性能出現在第 86 個訓練週期，其損失值為 3.9956。以下是模型訓練過程中的損失值變化趨勢圖以及實驗數據表格如圖(五)、圖(六)：

（插入損失值變化圖） 圖(四)



(圖四)

	Epoch	Learn rate	Avg Train loss	Train Accuracy	Avg Eval loss	Eval accuracy
1	0	0.0	6.189791270083535	0.0025592347348359994	6.187216567993164	0.0031494140625
2	1	.00010236998709491225	4.632840306278789	0.060518425021562156	4.632741117477417	0.0576904296875
3	2	0.0002047399741898245	4.198508667050775	0.07597679943609156	4.134325623512268	0.0786376953125
4	3	0.0003071099612847368	4.087874266474727	0.07707217331250983	4.120474410057068	0.080615234375
5	4	0.000409479948379649	4.056103768609082	0.07909583157951922	4.130804014205933	0.0748779296875
6	5	0.0005118499354745613	4.061045350881973	0.07837879657745361	4.034765720367432	0.0766845703125
7	6	0.0006142199225694736	4.048572855191019	0.07828412776488697	4.025387620925903	0.0762451171875
8	7	0.0006900287587898739	4.053008304114228	0.07706530055240966	4.069311285018921	0.070947265625
9	8	0.0006454628006031465	4.051074641556463	0.07682475363740335	4.036170434951782	0.0745849609375
10	9	0.000608548164413527	4.050239804254864	0.0788703233118045	4.058692741394043	0.083203125

圖(五)

92	91	.00019137954395089717	4.022752508368509	0.07974998038198558	4.058288812637329	0.079052734375
93	92	.00019033659589229245	4.029068343875351	0.07972404538135683	4.072228860855103	0.0744384765625
94	93	.00018931051512410888	4.019133406694432	0.08002251286199476	4.039987277984619	0.0763916015625
95	94	.00018830085184437035	4.031858798586875	0.0788520209542 1101	4.083324933052063	0.0727294921875
96	95	.00018730717286683074	4.023693176666624	0.07937680861241989	4.09150013923645	0.0774658203125
97	96	.00018632906084005817	4.026121288029407	0.07914625836771502	4.083125281333923	0.076953125
98	97	.00018536611351091333	4.02466181846202	0.07956622730393866	4.048500728607178	0.084716796875
99	98	0.00018441794302947 04	4.026228251717603	0.07919023509547368	4.068433690071106	0.081494140625
100	99	.00018348417529265247	4.02265363219655	0.0789698513142078	4.0509308815002445	0.073779296875
101	100	0.00018256444932405 81	4.0272210222055485	0.07913218901425906	4.011117792129516	0.0786865234375

圖(六)

從圖中可以觀察到：

初始階段損失值從 6.0 快速下降至 4.0，表明模型在早期階段有效學習了數據特徵。

在隨後的訓練過程中，損失值逐漸穩定，模型收斂效果良好。

模型參數設置

訓練時使用的參數設置如下：

批量大小（batch_size）：2

最大序列長度（max_sequence）：2048

Transformer 層數（n_layers）：6

多頭注意力機制（num_heads）：8

模型維度（d_model）：512

前向傳遞維度（dim_feedforward）：1024

丟棄率（dropout）：0.1

這些設置確保模型能夠在資源受限的情況下進行有效的訓練。

四、專題學習歷程

在專題的研究過程中，我們從以下幾個階段逐步完成了專題內容：

1. 找老師與確定研究方向

專題初期，我們首先尋求合適的指導老師，並通過多次討論，確定了研究的主題方向—基於 Music Transformer 的音樂生成模型。

2. 實驗室初步實踐與器材處理

我們作為實驗室的 Student Assistant (SA)，我們負責管理和處理實驗室的相關器材，從實際操作中熟悉了實驗環境的基本要求，這為後續的實驗奠定了基礎。

3. 學習 Linux 系統

由於實驗需要運行在 Linux 環境下，我們從基礎指令學起，逐步熟悉了 Linux 系統的架構與操作，完成了環境的部署與程式的執行。

4. 深度學習技術的學習與應用

我們深入學習了深度學習的相關理論與實踐，包括 Transformer 模型的基本結構及其在音樂生成中的應用。過程中，我們熟悉了 Pytorch 框架，並完成了 Music Transformer 的部署與測試。

5. 專題口試的準備與經歷

在中期口試階段，我們製作了簡報，整理了模型的訓練參數、生成樣本及其效果，並回答了評審老師對於模型選擇與結果的相關問題。口試的過程幫助我們更清晰地理解專題的目標與價值。

6. 調整實驗參數與優化生成結果

根據口試的建議，我們對模型參數進行了多次調整，包括：

num_prime 的設置（1 與 256 的對比）。

音樂生成長度參數（如 target_seq_length 和 max_sequence）。

通過不斷的測試與優化，我們最終生成了滿足專題目標的音樂樣本，並分析了

不同參數對生成效果的影響。

五、 結論與未來展望

5.1 結論

本專題成功實現了基於 Music Transformer 的音樂生成系統，並在模型架構中引入了相對位置表示（RPR）技術以提升生成效果。在訓練過程中，模型的損失值穩定於 3.995 表明生成的音樂片段在旋律的和諧性與多樣性上表現良好，能有效應用於遊戲配樂、教育工具等場景。

在研究過程中，我們完成了以下目標：

基於 Maestro V2 數據集進行高效數據預處理與模型訓練。

優化 Music Transformer 模型結構，解決長序列生成的依賴問題。

評估生成樣本的品質，驗證了模型在實際應用場景中的可行性。

5.2 未來展望

儘管本專題已取得一定成果，但仍存在一些需要進一步探索的方向：

數據集擴展與多樣化：

未來可引入更多元化的音樂數據集，例如流行音樂、古典音樂等，以提升模型對多種音樂風格的適應能力。

生成方法優化：

探索固定長度音樂生成策略，進一步提升生成旋律的結構性與穩定性。

多模態生成：

結合音頻、歌詞等多模態數據，實現更具創意性的音樂生成應用。

實時生成應用：

優化模型的計算效率，使其能夠支持即時音樂生成，拓展在現場表演與互動娛

樂中的應用可能性。

參 考 文 獻

1. MUSIC TRANSFORMER: GENERATING MUSIC WITH LONG-TERM STRUCTURE Cheng-Zhi Anna Huang* Ashish Vaswani Jakob Uszkoreit Noam Shazeer Ian Simon Curtis Hawthorne Andrew M. Dai Matthew D. Hoffman Monica Dinculescu Douglas Eck December 2018 (available online at https://arxiv.org/abs/1809.04281)
2. Self-Attention with Relative Position Representations (RPR) (Shaw et al., 2018) https://arxiv.org/pdf/1803.02155

附　錄　一

Clone 下來的深度學習網站

https://github.com/gwinndr/MusicTransformer-Pytorch/tree/master

數據預處理的參考網站

https://github.com/jason9693/midi-neural-processor

靜宜大學

資訊工程學系

專題題目

西元二〇二四年

十二月