# Easy Tunes: Simplified Music Generation Using Transformers

Michael Ingram

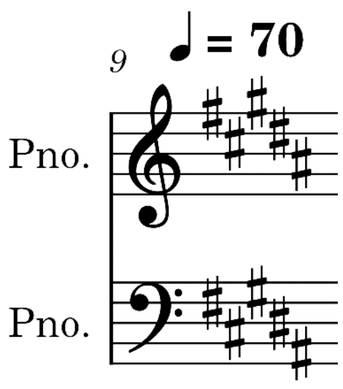
## Abstract

Modern solutions to music generation via deep learning are very **powerful, but expensive**, requiring high-end graphics cards and long run times. **Easy Tunes aims to reduce the complexity of these models, but still produce decent music**, with the aim of generating music on consumer devices.

## Background

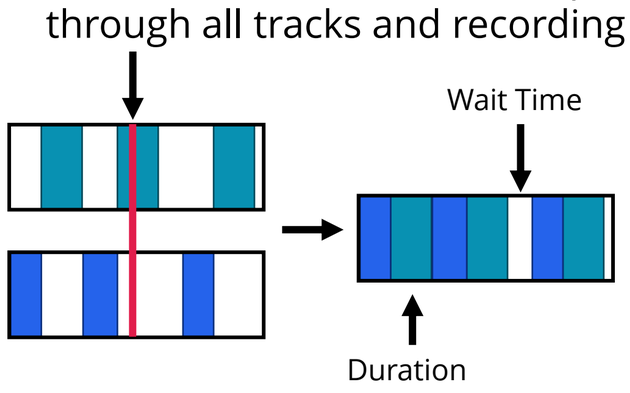
### Dataset

Almost **3,000 MIDI files** were downloaded from Ambrose Piano Tabs. These midi files were written specifically for **piano and keyboard**, which will substantially decrease model complexity.



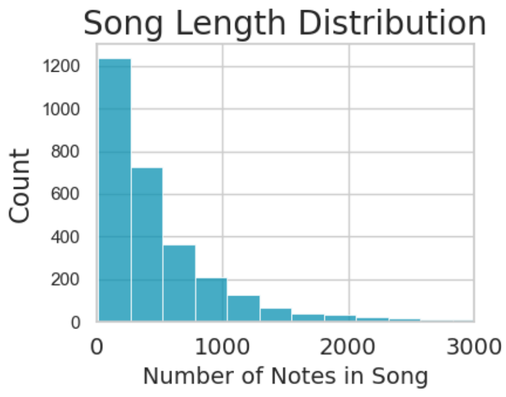

MIDI files are binary representations of songs which consist of tracks of messages with instructions as shown below. **MIDI messages contain extra information** such as time and key signatures, tempo, etc., which we can ignore to simplify the model.

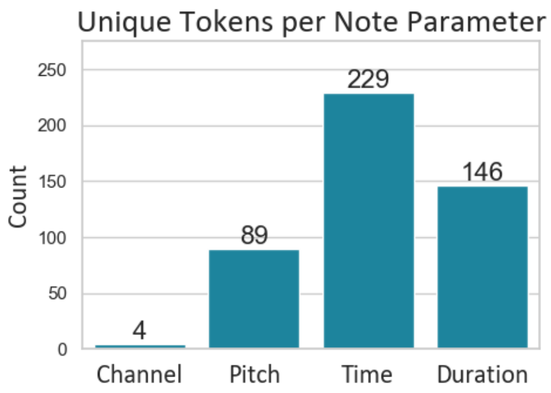
We distilled notes down to 4 parameters by running through all tracks and recording events as they occur.




| MIDI | Input |
|---|---|
| On/Off | Time |
| Time | Duration |
| Channel | Channel |
| Pitch | Pitch |
| Volume | |

To get a MIDI file out, we need to reverse this process with the model's output.

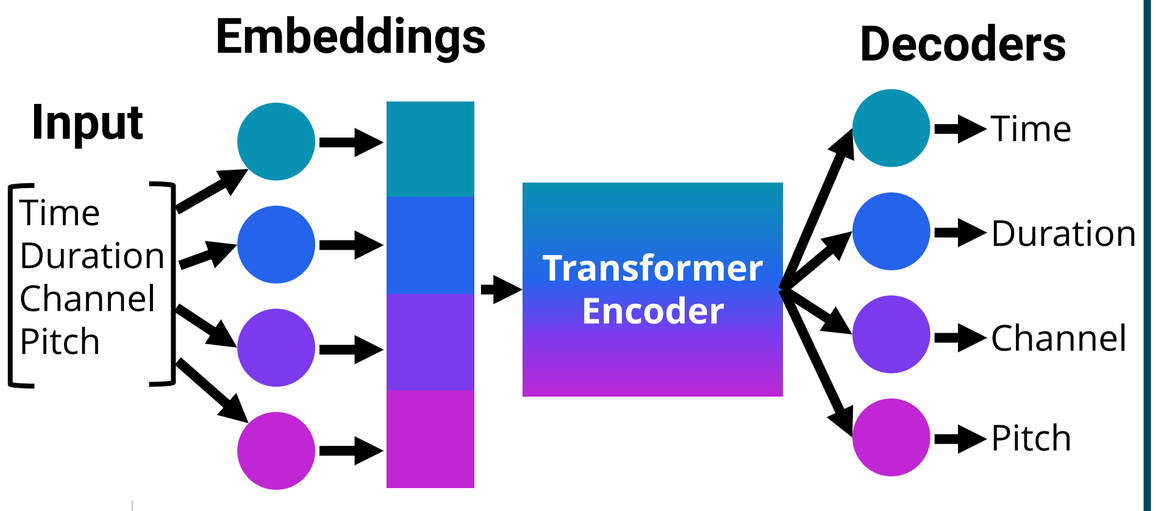


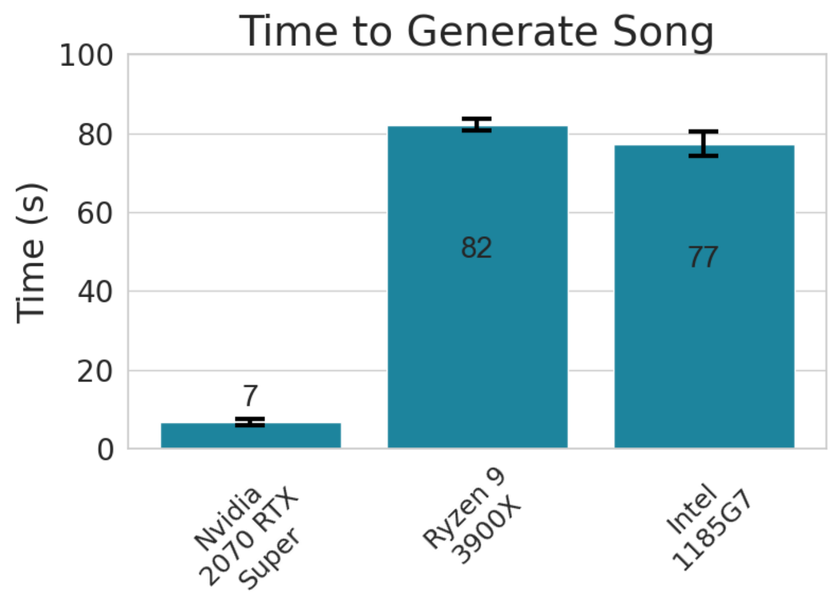

### Model & Embeddings

We used a transformer architecture with **learned embeddings for each of the important note parameters above**. MIDI files record 127 different notes, and there are hundreds of possibilities for timings and durations of notes. To tokenize all of that **would require over a million unique tokens**.

Instead, we **tokenized the note parameters independently** and added embeddings for each of them separately.



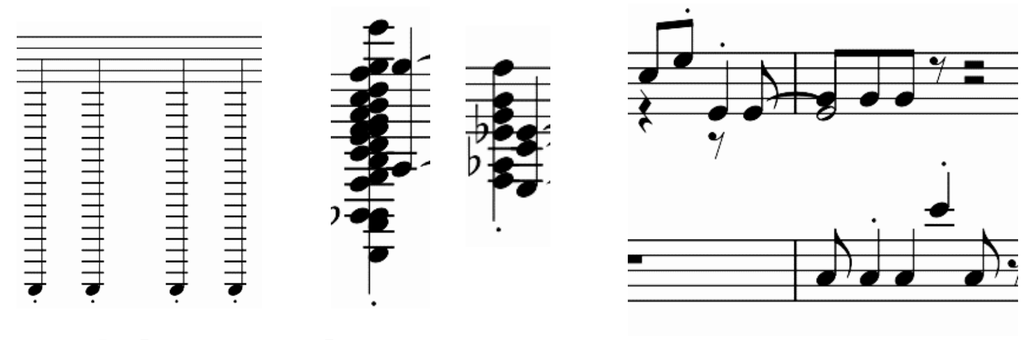

## Results



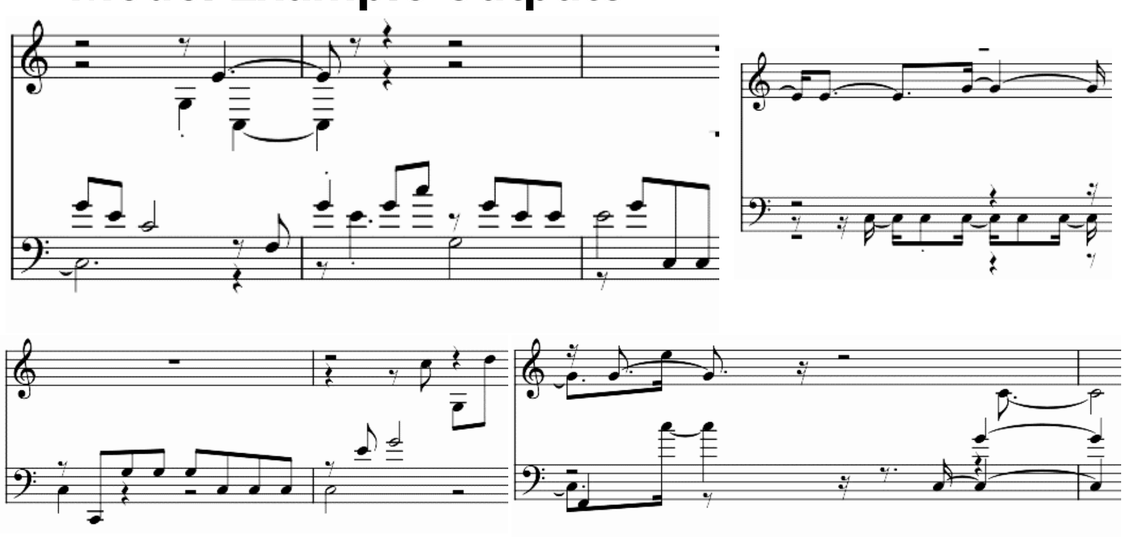

Music was generated using **three different devices**: RTX 2070 Super Graphics Card, Ryzen 9 3900X desktop CPU, and an Intel i7 11th generation mobile CPU. **The average run time across 10 prompts was recorded**. The average time to generate a song using normal consumer CPU's was around **a minute and a half**.

Some examples of music generated by the model are shown below:

### Learning Experiences



### Model Example Outputs



## Conclusions

While the music generated wasn't as complex as music from state-of-the-art models, **Easy Tunes was able to generate very simple music on consumer grade hardware in a short amount of time**.

### Next Steps

Longer sequence sizes drastically slowed down the model, because of the transformer's $O(n^2)$ performance. **Other architectures**, like the Sparse Transformer $O(n\sqrt{n})$ or Mamba $O(n)$ may perform better.

Because the model generates quickly, **adding back some features such as the metadata and volume** may improve predictions and musicality, and **adding more complexity** will allow for the model to learn more patterns.

Finally, using a **better dataset** or **fine tuning** on a particular genre will allow users to get the exact style of music they want