1 Methodology

We outline and detail the steps taken to execute this project from beginning to end.

1.1 Designing the Function

Consider the biological process of hearing a sound wave and matching it to a source. We can model this behavior by some unknown function F. We produce an approximation of that function F^* that can map the contents of a sound file that have been generated by a chaotic synthesizer to a potential source. For a set of inputs $\vec{x} = \{x_0, x_1, x_2, ..., x_{p-1}\}$ and a set of classes 0 through $k-1$, we denote this function as:

$$F^* : \vec{x} \rightarrow \{0, 1, 2, ..., k-1\} \tag{1}$$

1.2 Collecting and Pre-processing Raw Data

In order to train the Multimodal neural network to identify musical instrument sources, we need a suitable data set to present to the model. University of Iowa Electronic Music Studio, and the London-Based Philharmonia Orchestra each have a large collection of publicly available audio files Citations!. These contain short segments of musical instruments performing a single note or a collection of notes in succession.

To ensure that these data sets are roughly homogeneous, we read each sample from it's original format, *.aif*, *.mp3*, or similar, and rewrite each sample as a new *.wav* files, sampled at 44.1 kHz, with a 16 bit depth check this!. This ensures that all data will have a consistent format when features are extracted. We also use this stage to ensure that each audio file has a correct label, and to determine the number of unique output classes.

1.3 Designing Classification Features

The performance of a neural network is largely dependent of designing an appropriate set of classification features. These are properties of wave forms that can be represented by a numerical value or several numerical values and are used as the primary tool in classification. There are used in place of a full waveform to represent a file's contents We use tools from physics, mathematics, and signal processing to define and explore a comprehensive set of features that enables a high performance of the classifier.

1.4 Designing A Complementary Network Architecture

With the appropriate set of features designed, and the number of output determined, we can organize the structure of the neural network function. We construct a multimodal neural network than processes two input arrays derived from the same audio sample that share a common label. This network is designed to handle and process each respective input independently, and concatenate the results to produce a single output.

1.5 Testing and Evaluating Network Performance

We divide the raw data set up into a training and testing sets. We employ cross-validation and compute the results of performance metrics to ensure that out model is making reliable predictions, and generalize appropriately. In this stage, we also choose the value of hyper-parameters, activation functions and layer widths to best compliment the chosen features. This process is repeated and expanded upon until we have produced a model with a sufficient performance.

1.6 Running Predictions of Chaotic Synthesizer Files

Once we have established a suitable performance of the model, we allow the model to run predictions on the un-labeled chaotic synthesizer wave forms. We output the prediction results to a file, and compare the neural network predictions against human predictions. If further corrections are needed, we revert and re-deign the features, architecture, or hyper-parameters as needed.

References

[1] Geron, Aurelien. *Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly, 2017.

[2] Geron, Aurelien. *Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. 2nd ed., O'Reilly, 2019.

[3] Goodfellow, Ian, et al.*Deep Learning*. MIT Press, 2017.

[4] James, Gareth, et al. *An Introduction to Statistical Learning with Applications in R*. Springer, 2017.

[5] Khan, M. Kashif Saeed, and Wasfi G. Al-Khatib. "Machine-Learning Based Classification of Speech and Music." Multimedia Systems, vol. 12, no. 1, 2006, pp. 55–67., doi:10.1007/s00530-006-0034-0.

[6] Levine, Daniel S. *Introduction to Neural and Cognitive Modeling*. 3rd ed., Routledge, 2019.

[7] Liu, Zhu, et al. "Audio Feature Extraction and Analysis for Scene Segmentation and Classification." Journal of VLSI Signal Processing, vol. 20, 1998, pp. 61–79.

[8] Loy, James , *Neural Network Projects with Python*. Packt Publishing, 2019

[9] McCulloch, Warren S., and Walter Pitts. "A Logical Calculus of the Ideas Immanent in Nervous Activity." *The Bulletin of Mathematical Biophysics*, vol. 5, no. 4, 1943, pp. 115–133.

[10] Mierswa, Ingo, and Katharina Morik. "Automatic Feature Extraction for Classifying Audio Data." *Machine Learning*, vol. 58, no. 2-3, 2005, pp. 127–149., doi:10.1007/s10994-005-5824-7.

[11] Mitchell, Tom Michael. Machine Learning. 1st ed., McGraw-Hill, 1997.

[12] Olson, Harry E. *Music, Physics and Engineering*. 2nd ed., Dover Publications, 1967.

[13] Peatross, Justin, and Michael Ware. *Physics of Light and Optics.* Brigham Young University, Department of Physics, 2015.

[14] Petrik, Marek. "Introduction to Deep Learning." Machine Learning. 20 April. 2020, Durham, New Hampshire.

[15] Short, K. and Garcia R.A. 2006. "Signal Analysis Using the Complex Spectral Phase Evolution (CSPE) Method." AES: *Audio Engineering Society Convention Paper.*

[16] Virtanen, Tuomas, et al. *Computational Analysis of Sound Scenes and Events.* Springer, 2018.

[17] White, Harvey Elliott, and Donald H. White. *Physics and Music: the Science of Musical Sound*. Dover Publications, Inc., 2019.

[18] Zhang, Tong, and C.-C. Jay Kuo. “Content-Based Classification and Retrieval of Audio.” *Advanced Signal Processing Algorithms, Architectures, and Implementations VIII*, 2 Oct. 1998, pp. 432–443., doi:10.1117/12.325703.