

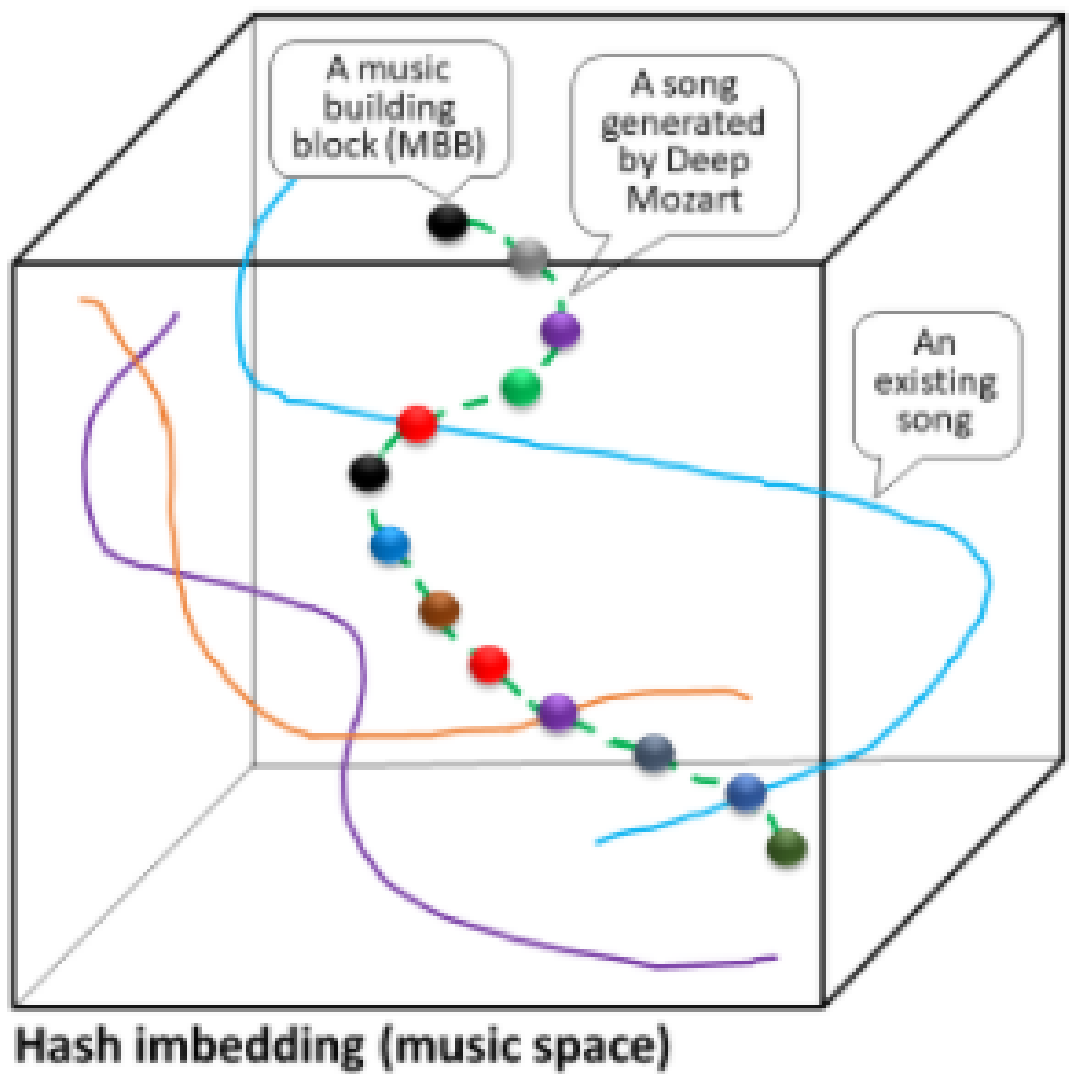
Deep Composer: A Hash-Based Duplicative Neural Network For Generating Multi-Instrument Songs

Jacob E. Galajda Brandon Royal Dr. Kien A. Hua

University of Central Florida, Department of Computer Science

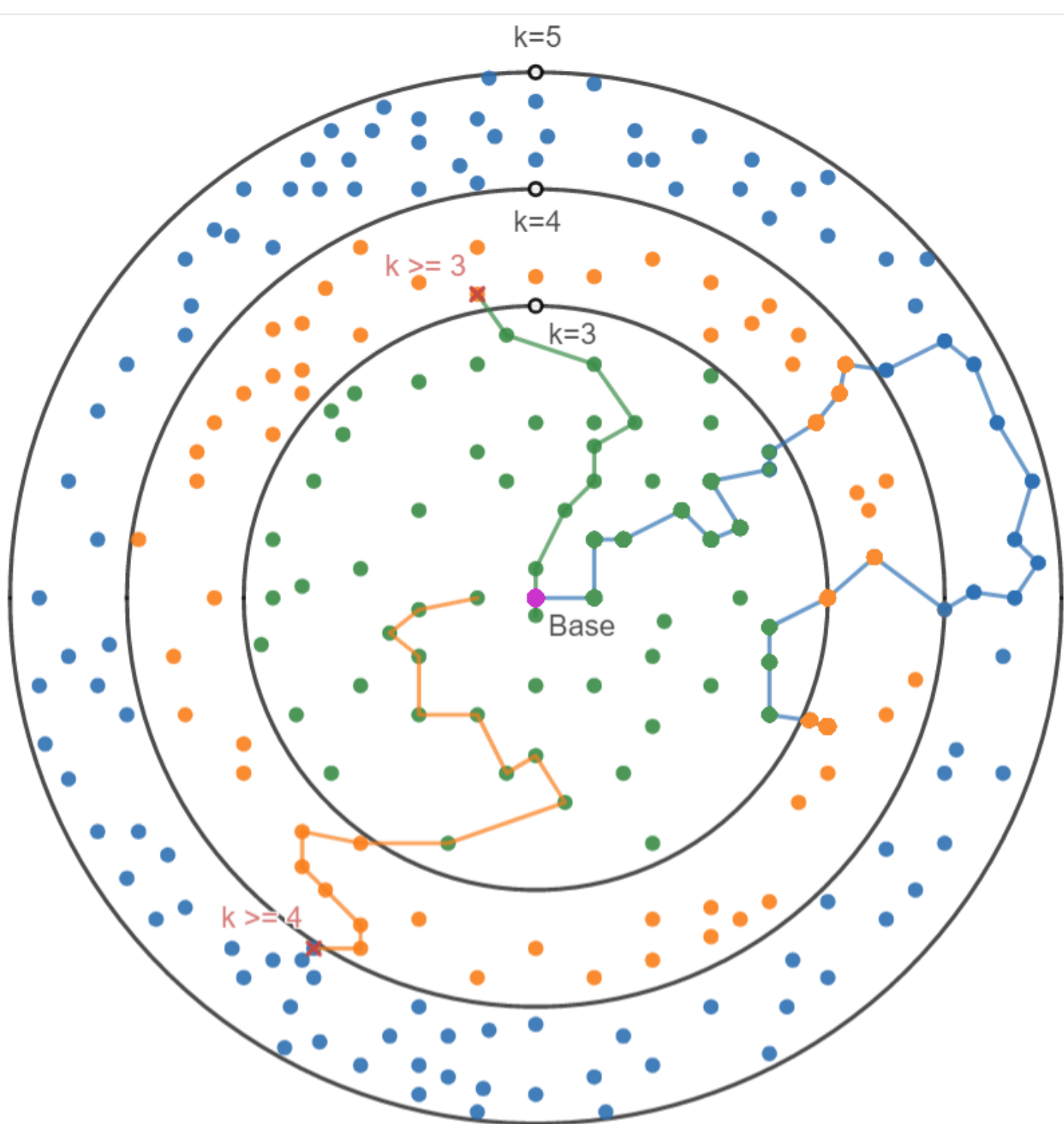
Intelligence Duplication

Deep Composer utilizes a music generation model based on database query processing. The music database may include hundreds of thousands, or more, of music building blocks (MBBs). These MBBs are tiny music segments obtained from existing songs, each a few seconds in duration. This intelligence is acquired through the learning from the songs selected from the music database. Visually, we can view each MBB as a feature point in a music space (i.e., the hash embedding).



Deep Composer also has the potential to offer great composers from different eras the opportunity to collaborate by using MBBs from their music. This is in fact the scenario selected for the performance study in this paper, except with many composers. In this framework, the music space is a hash embedding capable of capturing how these composers would compose their music. Their individual musical styles take turns to play the music in the constructed song. This capability can also be applied to learn and compose a truly worldwide fusion genre of music.

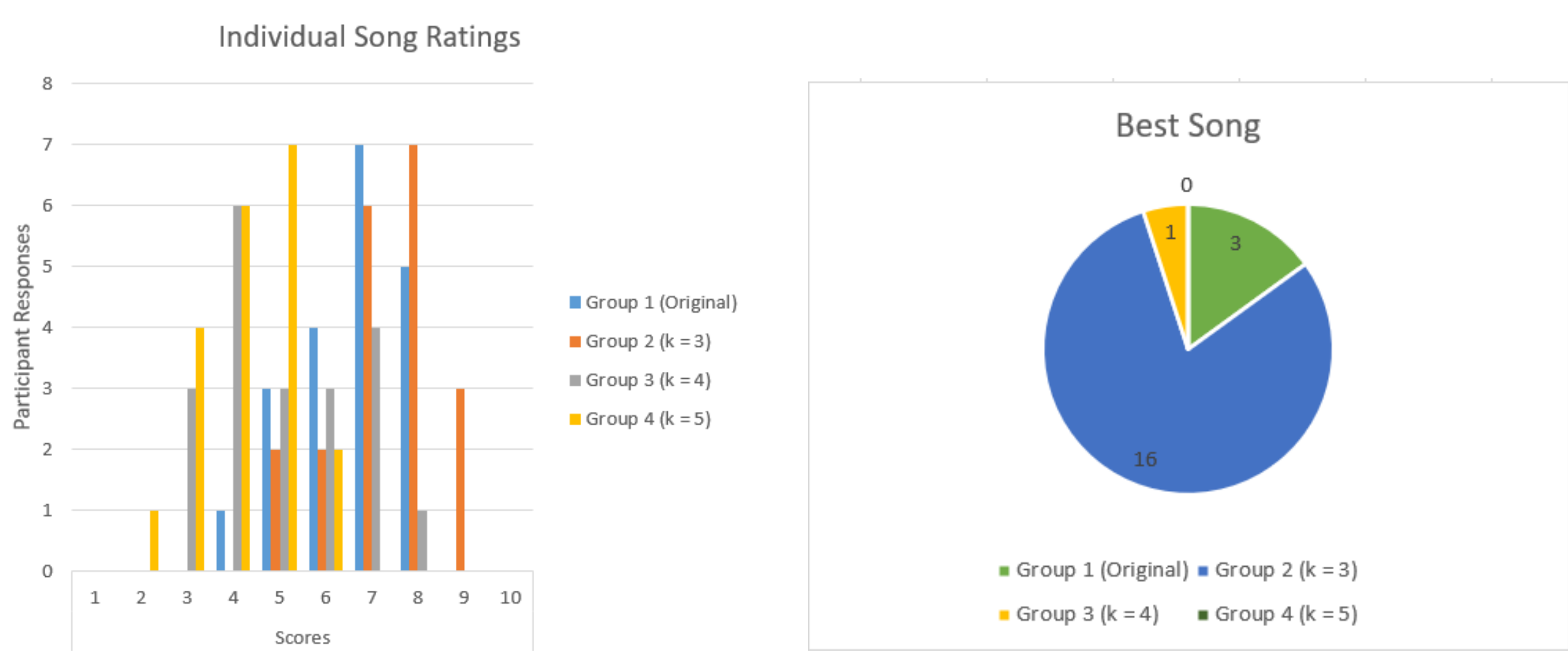
Segment Barrier Retrieval



the origin and any next segment are used to detect segment sparsity. To summarize, the retrieval branch is closed off and another branch is created near the previous origin when segment sparsity is detected.

Segment Barrier Retrieval (SBR) is a simple and efficient algorithm that prevents retrievals from branching too far away from areas of high density in the hash-space. The algorithm's ability to detect segment sparsity is dependant upon the set Hamming distance threshold denoted by k . We define an origin segment at the beginning of each retrieval branch. The Hamming distance between

Survey Results



Using a large compilation of classical string quartet dataset, consisting of Mozart, Chopin, Beethoven and many more, we trained one network and developed four different groups with various k values. The best songs (ie. the song with the least segment sparsity) from each group. Group 1 was our baseline, and Groups 2-4 had set k values of 3-5, respectively. We surveyed students from the University of Central Florida with decent knowledge of music to score our each group's performance.

Research Summary and Future Works

In conclusion, Deep Composer generates a unique, *custom* embedding filled with hash-pair encodings instead of features or single hash values. This hash embedding generated by the network extends Deep Composer's capacity, giving it the ability to generate a potentially infinite number of songs without additional training. Deep Composer places segments based on their composability in which the network learns how to duplicate the actual composer(s) rather than replicate the globalized genre of music. To our knowledge, this implementation is unlike any other network that has been studied.

For these reasons, we introduce a new field of study, Intelligence Duplication, to explore this subject further. AI research generally focuses on developing techniques to mimic universal intelligence. Intelligence Duplication (ID) research focus on techniques to artificially duplicate/clone a specific mind such as Mozart to perform tasks only Mozart could. In this research, we attempt Intelligence Duplication, in which the learning is not to capture general knowledge to solve general problems, but to duplicate/clone a specific mind such as "recreating" Mozart. More specifically, we use music as a model to illustrate ID with Mozart in future works.