DJ AI: Optimizing Playlist Alignment and Generating Transitions with Generative and Embedding Models

ABSTRACT

Digital music platforms have transformed the listening experience through curated playlists and transitions, yet many transition creating systems primarily serve professional DJs with many manual features to be set while overlooking amateur performers and everyday listeners. In this study, we introduce DJ-AI, a novel framework that bridges this gap by analyzing detailed musical features to optimize song sequences and create harmonic transitions between those songs. Our approach employs graph based optimization techniques to efficiently arrange playlists by mapping song relationships and determining the best transition paths. Additionally, we integrate MusicGen—a generative model for generating coherent musical continuations—and MERT audio embedding model, which capture nuanced musical attributes, to enhance the smoothness of transitions. Experimental evaluations reveal that DJ-AI outperforms traditional crossfade methods in generating smooth and coherent transitions. This framework paves the way for AI-driven adaptive mixing solutions, making seamless music transitions more accessible to a broader audience.

KEYWORDS

Music transitions, Playlist automation, AI-powered DJ

ACM Reference Format:

. 2025. DJ AI: Optimizing Playlist Alignment and Generating Transitions with Generative and Embedding Models. In *Proceedings of 20th Audio Mostly Conference (Audio Mostly 2025)*. ACM, New York, NY, USA, 5 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The evolution of music consumption has been nothing short of transformative. Historically, music enthusiasts manually sequenced vinyl records, relying on their intuition and musical knowledge to create a coherent listening experience. With the advent of digital technologies, the art of playlist curation transitioned to automated systems that primarily consider listening history, often neglecting the rich tapestry of musical attributes such as tempo, harmony, and rhythm. Similarly, while DJ applications have advanced in providing real time transition effects, they usually emphasize technical mixing rather than an integrated approach that accounts for the musical compatibility of tracks. This dichotomy has resulted in systems that serve professional DJs or passive listeners but fall short for the diverse needs of end users and amateur DJs.

Audio Mostly 2025, June 30 – July 4, 2025, Coimbra, Portugal

ACM ISBN 978-1-4503-XXXX-X/2025/06
https://doi.org/XXXXXXX.XXXXXXX

Recent advancements in artificial intelligence, sound analysis, and music generation have paved the way for a paradigm shift in how playlists are created and transitions are designed. Our proposed system, DJ-AI, leverages these technologies to redefine the music listening experience. Unlike traditional recommendation algorithms that prioritize user history and simple metadata, DJ-AI analyzes songs by delving into their intrinsic musical attributes, such as tempo, harmonic structures, and rhythmic patterns. This comprehensive analysis facilitates the generation of optimized song sequences that ensure both harmonic coherence and rhythmic continuity.

Central to our approach is the integration of advanced tools like MusicGen and MERT embeddings. MusicGen contributes by providing AI-driven sound synthesis capabilities, generating smooth and contextually appropriate musical transitions. In contrast, MERT embeddings capture nuanced musical features that are pivotal for selecting the best transition segments. This dual integration enables DJ-AI to employ techniques akin to those used by professional DJs while remaining accessible to non-experts.

Furthermore, our system draws an analogy to prompt engineering in large language models: just as the quality of structured prompts significantly influences the output, the ordering of songs based on musical features critically shapes transition quality. By imposing structured musical constraints, DJ-AI arranges tracks in an optimal, feature-wise sequence, ensuring that each song naturally flows into the next. This careful sequencing not only improves harmonic and rhythmic continuity but also enhances the overall auditory experience by facilitating smoother transitions. In essence, a well-ordered playlist serves as a refined prompt that drives the system's success, addressing a significant gap in current research where sequencing and transitions are typically treated as separate challenges.

2 RELATED WORKS

Early work in automated music transition and sequencing predominantly relied on classical signal processing and graph-based optimization techniques. For instance, Robinson and Brown [12] proposed an innovative method for generating audio crossfades in the time-frequency domain using graph cuts. Their approach discretizes the frequency spectrum into bins and formulates the crossfade as a min-cut problem, thereby enabling per-frequency seam selection that is more adaptable than standard amplitude-based crossfading.

Parallel to this, Bittner et al. [1] addressed the broader problem of automatic playlist sequencing and transitions. Their work casts the sequencing problem as a graph traversal task, where songs are represented as vertices weighted by acoustic and musical similarity. By solving for the shortest Hamiltonian path (or cycle) in the graph, they achieved coherent sequencing that respects key, tempo, and timbral continuity between tracks.

More recent advances have shifted towards deep learning techniques that can learn rich representations directly from audio. Hsu and Chang [5] introduced a transformer-based model for generating music transition sequences (MTS), leveraging an encoder-decoder architecture that captures fine-grained musical attributes such as tempo and harmonic progression. Unlike traditional crossfade methods, their model can synthesize transitions that are both musically coherent and dynamically adaptive.

Building on these advances, our work integrates two state-of-the-art models to further enhance transition generation. First, we incorporate the MERT model [10], which uses large-scale self-supervised training to produce detailed acoustic and musical embeddings. These embeddings are crucial for identifying optimal transition points between songs. Second, we leverage MusicGen [2], a controllable music generation framework that employs efficient codebook interleaving and conditional generation. MusicGen not only synthesizes high-fidelity audio transitions conditioned on both textual and melodic cues but also complements the analysis provided by MERT.

By combining the insights from classical graph-based methods with the representational power of modern deep learning models, our approach offers an end-to-end solution. It addresses both the sequencing and transition generation tasks in a manner that is accessible to non-professional users while retaining the sophisticated qualities typically produced by expert DJs.

3 METHODOLOGY

Figure 1 illustrates the process of DJ-AI transforming a standard playlist into a transition-enhanced playlist. The key phases of this method include analyzing songs, identifying optimal transition points, and generating transitions using MusicGen. Initially, musical attributes such as tempo and energy are extracted for each song, followed by compatibility analysis to determine the best playlist ordering. In the final stage, transition segments are identified, and smooth transitions are achieved through conditional music generation and audio processing.

3.1 Feature Extraction

Feature extraction is a fundamental step in playlist optimization, as it involves obtaining the musical attributes that define the compatibility between songs. In this approach, the tonal key, tempo, and energy were extracted from audio files.

The tonal key was determined using chroma features, which calculate the intensity of each note class. The Krumhansl-Schmuckler key-finding algorithm matched the extracted chroma profiles with major and minor key templates to identify the musical key of each track [8]. These keys were then converted into the Camelot Wheel format to facilitate harmonic mixing, optimizing transitions between harmonically compatible keys [4].

Compatibility analysis was performed by comparing extracted features. Two songs were considered compatible if they met the following criteria:

- Camelot keys must be identical, adjacent, or harmonically complementary.
- Tempo difference should not exceed ±2 BPM.
- Energy level should generally increase but should not exceed a variation of 0.2 RMS units.

3.2 Playlist Optimization and Evaluation

A graph-based approach was used to determine the optimal song sequence. Each song was represented as a node in a directed graph, with directed edges connecting nodes that met compatibility criteria.

A Depth-First Search (DFS) algorithm was applied to explore all possible transition paths, and the longest paths were selected to represent the most coherent playlists. This technique shares similarities with the Traveling Salesman Problem (TSP), which seeks to find an optimal route through a set of points [6].

The quality and coherence of generated playlists were assessed using cosine similarity between song embeddings, ensuring the selection of the most optimal playlist order. Embeddings were derived using audio features such as MFCCs, chroma features, spectral contrast, and more.

3.3 Optimal Transition Segment Detection and Generation

The transitions created by DJ-AI leverage MusicGen's conditional music generation capability to synthesize transition appropriate audio [2]. MusicGen's ability to generate audio while maintaining melodic alignment was a key factor in its selection. This approach was further enhanced using MERT embeddings, which were trained with models incorporating both acoustic and tonal features, ensuring rich musical context [10].

To determine the most compatible segments between two songs, the last 30 seconds of each track were divided into overlapping 10 second segments using a 2 second sliding window. This segmentation enabled a detailed analysis of features such as MFCCs, spectral centroid, spectral bandwidth, chroma, energy, and tempo. These features were then combined with MERT embeddings to enhance musical representation, ensuring the best possible transition quality. Prior studies have demonstrated that combining features improves embeddings by adding complementary information without distorting the original representation [14, 15]. To balance feature effects, all features were normalized to zero mean and unit variance [14]. Segment similarity was computed using cosine similarity.

Beat Matching for Rhythmic Continuity: In addition to harmonic and spectral analysis, DJ-AI applies beat matching to ensure rhythmic consistency between consecutive tracks. Beat mismatches can cause jarring transitions, particularly in genres with strong rhythmic structures such as electronic dance music. To prevent this, DJ-AI synchronizes the beats of the transition and crossfade segments through the following steps:

(1) Detecting Beats: Using Librosa's beat tracking algorithm, beats are identified in both the transition and crossfade segments.
(2) Computing Tempo Differences: The tempo of each segment is estimated, and the ratio between them is calculated.
(3) Time Stretching for Alignment: The transition segment is time stretched to match the tempo of the crossfade segment using phase preserving algorithms.

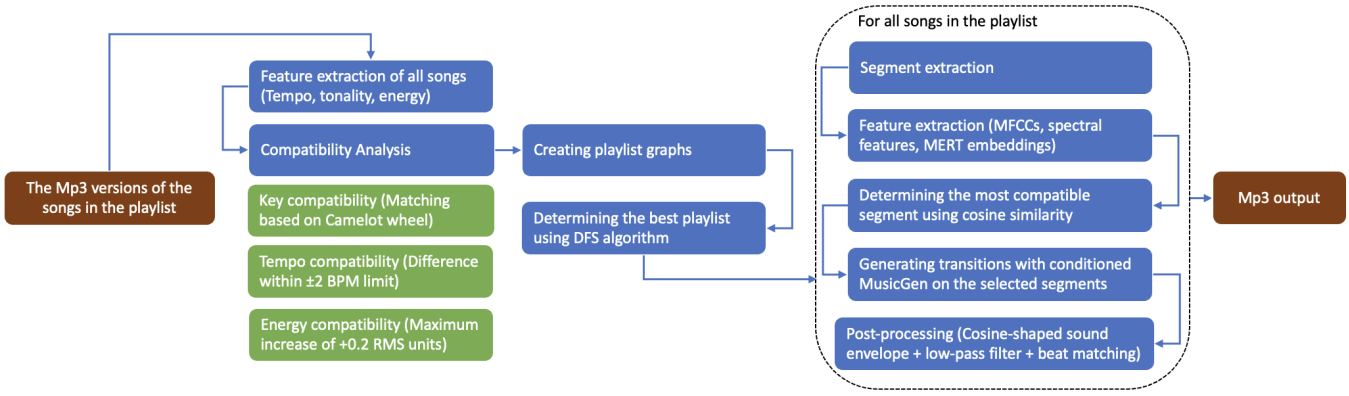

Figure 1: Transitioning from a Playlist to a DJ-AI Enhanced Playlist

(4) Aligning Beat Positions: The starting point of the transition segment is adjusted so that its first detected beat aligns with the closest beat in the crossfade segment.

To achieve phase preserving time stretching, DJ-AI employs a phase vocoder based approach, which operates in the frequency domain to modify tempo without altering pitch. This method first applies a short time Fourier transform (STFT) to convert the audio signal into a time frequency representation. The algorithm then adjusts the timing of spectral components while preserving phase relationships between frequency bins, preventing phase smearing and unnatural artifacts. Finally, the modified signal is reconstructed using an inverse STFT (ISTFT), ensuring that the stretched audio remains perceptually natural and rhythmically coherent.

Mathematically, the tempo adjustment is defined as:

$$\text{tempo\_ratio} = \frac{\text{tempo}_{\text{crossfade}}}{\text{tempo}_{\text{transition}}} \tag{1}$$

$$y_{\text{adjusted}}[n] = y_{\text{transition}}\left(\frac{n}{\text{tempo\_ratio}}\right) \tag{2}$$

where $y_{\text{transition}}$ represents the original transition audio, and y_{adjusted} is the beat matched transition segment.

Seamless Blending with Crossfade and Envelope Shaping: Once the most compatible segments were identified and beat-matched, a crossfade technique was applied to blend the transitions smoothly. To ensure smooth loudness transitions, a cosine-shaped volume envelope was applied:

$$V(n) = \begin{cases} \frac{1}{2}\left(1 - \cos\left(\frac{\pi n}{N_1}\right)\right), & 0 \le n < N_1 \\ \frac{1}{2}\left(1 + \cos\left(\frac{\pi(n-N_1)}{N_2-N_1}\right)\right), & N_1 \le n < N_2 \end{cases} \tag{3}$$

Additionally, MusicGen was used to generate synthetic transition audio. The model was conditioned on the linear blend of selected segments to produce transition appropriate sound. However, since the model tends to adhere closely to the input melody, the generated transition audio was twice the length of the intended crossfade duration, and only the latter half was used for the transition [2].

Low-Pass Filtering for Smoother Transitions: In the final step, a Butterworth low-pass filter was applied to remove high-frequency artifacts, further enhancing the transition quality:

$$H(s) = \frac{1}{\sqrt{1 + \left(\frac{s}{\omega_c}\right)^{2N}}} \tag{4}$$

where $\omega_c = 2\pi f_c$ is the cutoff angular frequency and N is the filter order. In digital implementation, the difference equation is:

$$y[n] = \sum_{k=0}^{M} b_k x[n-k] - \sum_{k=1}^{N} a_k y[n-k] \tag{5}$$

where $x[n]$ is the input and $y[n]$ is the output signal. This filtering process enhances the transition signal by removing unwanted noise and ensuring a smooth blending of audio tracks.

By integrating harmonic aware segment selection, beat-matching adjustments, envelope shaping, and MusicGen based synthesis, DJ-AI produces rhythmically and melodically seamless transitions that outperform conventional crossfade methods. These techniques ensure that playlist automation can generate transitions that feel intentionally mixed, rather than mechanically overlapped, significantly improving user experience.

4 EXPERIMENTS AND RESULTS

Selecting appropriate evaluation metrics is a critical step in ensuring the quality and coherence of transitions used in the DJ-AI project. Since transitions between tracks require both harmonic and rhythmic compatibility, various commonly used music analysis metrics were examined, and those best suited to the system's requirements were selected. The chosen metrics include chromatic cosine distance and Dynamic Time Warping (DTW), as they provide the most balanced evaluation set considering tonal, rhythmic, and perceptual aspects.

4.1 Evaluation Metrics Analysis

First, various metrics widely used in the literature were analyzed for their suitability in evaluating DJ transitions. Mel-Cepstral Distortion (MCD) is a method that measures the spectral distance between audio signals using Mel-Frequency Cepstral Coefficients

(MFCCs). While effective for assessing timbral similarity, it was deemed insufficient for evaluating musical transitions as it does not account for rhythmic or harmonic compatibility [9].

Dynamic Time Warping (DTW) aligns time-dependent sequences by minimizing time distortions, making it an effective method for rhythmic compatibility. However, its computational cost presents a disadvantage for real-time applications [13].

Fréchet Audio Distance (FAD) is adapted from the Fréchet Inception Distance (FID) method and can evaluate perceptual audio quality without requiring a reference. By measuring tonal and timbral consistency, it provides a useful approach for evaluating the subjective aspects of music transitions [7]. However, since MERT is used for selecting appropriate transition segments in this study, FAD's reliance on embeddings could introduce bias into the evaluation, as MERT embeddings inherently shape the system's decision-making process.

Beat-Synchronized Chromatic Distance, which evaluates harmonic and rhythmic similarity by combining chromatic features with beat-synchronized analysis, effectively normalizes tempo variations. However, it requires cross-correlation of chromatic matrices for beat alignment, making it computationally intensive [3].

After analyzing the trade-offs between computational efficiency, relevance, and applicability, cosine distance and DTW were determined to be the most balanced metric set for evaluating transitions in this system.

4.2 Selected Metrics and Implementation Process

In this study, cosine distance and Dynamic Time Warping (DTW) were used to evaluate transition quality. Cosine distance was chosen to measure harmonic similarity and was computed between chromatic feature vectors extracted from audio files. Lower cosine distance values indicate stronger tonal compatibility between two tracks.

For rhythmic alignment, DTW was employed to determine the optimal time alignment between two segments while preserving rhythmic continuity in transitions. A Manhattan distance-based local cost function was used to balance computational complexity [11, 13].

The experiment consisted of two phases:

1. **Curated Playlist Transition Test:** In this phase, the system's transition performance was analyzed using a carefully selected playlist of harmonically compatible tracks. The playlist was constructed using the methods outlined in the Playlist Optimization section. The goal was to measure whether the system successfully optimized transition segments to create seamless transitions.

2. **Random Song Transition Test:** In this phase, the same tests were conducted using randomly selected songs to evaluate the system's performance when handling transitions between tracks without harmonic or tempo compatibility.

In both phases, two transition methods were tested:

(1) **DJ-AI Method:** After identifying the most compatible segments of both tracks, a transition was generated.
(2) **Crossfade Method:** A traditional crossfade was applied, merging the last 15 seconds of the first song with the first 15 seconds of the second song without any optimization.

During these tests, chromatic cosine distance and chromatic DTW distances were computed for each transition method. In the curated playlist test, the system's performance in terms of tonal coherence and rhythmic alignment was evaluated. In the random song test, the importance of transition optimization and segment selection was analyzed.

4.3 Evaluation Process

The evaluation process began with extracting chromatic and rhythmic features using music information retrieval libraries. A sliding window approach was then applied to encode transition segments for analysis. All possible transitions were systematically scored based on the selected metrics, and the obtained results were validated against subjective evaluations from human listeners. The final measurements served as the foundation for assessing transition quality and system performance.

4.4 Key Findings and Results

The evaluation process compared two different transition methods. The first method, DJ-AI, selected the most compatible songs from a large playlist, identified optimal transition points between them, and generated a transition. The second method, crossfade, applied a traditional static transition by directly blending the last 15 seconds of the first song with the first 15 seconds of the second song.

During the experiment, since the curated playlist was optimized for compatibility, the differences between the two methods were relatively small. However, when using randomly selected songs, the advantages of DJ-AI became more evident. This result demonstrates that the system's optimization, which considers tonal, tempo, and energy factors, has a direct impact on transition quality.

Evaluation Metric	DJ-AI	Crossfade
Chromatic Cosine Distance	0.3735	0.3888
Chromatic DTW Distance	4.3117	4.5018

Table 1: Test results using a curated playlist (Lower is better)

As seen in Table 1, a slight difference in chromatic cosine distance was observed between the two methods. This indicates that both methods achieved similar tonal transition quality. However, the chromatic DTW distance was lower for DJ-AI, demonstrating its ability to better align transitions rhythmically.

These results highlight that transition quality is directly related not only to tonal compatibility but also to rhythmic alignment. In DJ sets and automated playlist systems, selecting optimal transition segments leads to a smoother listening experience compared to static crossfades.

Evaluation Metric	DJ-AI	Crossfade
Chromatic Cosine Distance	0.3894	0.4112
Chromatic DTW Distance	4.4038	4.7111

Table 2: Test results using random songs (Lower is better)

Table 2 presents the evaluation results for transitions using randomly selected songs. In this experiment, where no playlist optimization was applied, both chromatic cosine distance and chromatic DTW distance were significantly higher than in the curated playlist tests.

This confirms the effectiveness of playlist optimization: when compatible songs are selected, transition scores improve, leading to smoother transitions. Additionally, even with random songs, DJ-AI outperformed the traditional crossfade method, demonstrating the importance of selecting the best transition segments and the effectiveness of the system in optimizing transitions.

Notably, the chromatic DTW distance improved from 4.7111 (random songs) to 4.5018 (playlist-optimized songs) and further down to 4.3117 when the best transition segments were selected. Similarly, chromatic cosine distance improved from 0.4112 (random songs) to 0.3888 (optimized songs). These findings indicate that DJ-AI provides better transitions than traditional crossfade methods, even for randomly selected tracks.

4.5 Human Evaluation

To assess the perceived quality of DJ-AI's transitions further, we conducted a blind listening test where participants compared DJ-AI transitions with traditional crossfade transitions. The objective was to evaluate which method was preferred when participants were unaware of the underlying technology. A total of 20 participants, including professional DJs, amateur DJs, and casual listeners, were recruited. Each participant listened to ten transition pairs, each containing a transition generated by DJ-AI and a baseline crossfade transition with a fixed 10 second overlap. The order of the two transitions was randomized for each participant to prevent bias. After listening to each pair, participants were asked to select the transition they preferred without knowing which was DJ-AI.

The results of this blind test indicate that DJ-AI transitions were chosen **70.5%** of the time, while the baseline crossfade was preferred in only **29.5%** of cases. This strong preference for DJ-AI suggests that transitions generated through harmonic aware AI techniques provide a more seamless and musically coherent listening experience compared to conventional crossfading methods. Many participants described the DJ-AI transitions as feeling more natural and integrated. Others noted that crossfade felt static while DJ-AI felt more dynamic. These findings reinforce the effectiveness of AI-driven transition modeling, particularly in improving the continuity of automated playlists. The experiment demonstrates that a structured approach can significantly enhance the listener experience by producing blends that closely resemble professional DJ mixing.

5 CONCLUSION

In this study, we introduced DJ-AI, an integrated framework designed to optimize playlist sequencing and generate harmonic transitions using both audio processing techniques and state-of-the-art models. By combining graph based playlist optimization, detailed audio feature extraction, segment level similarity assessments with MERT embeddings, and conditional music generation with MusicGen, DJ-AI achieves seamless transitions that rival those produced by professional DJ tools. Through rigorous evaluations on curated and random playlists, we demonstrated that DJ-AI outperforms traditional crossfade approaches in both harmonic coherence and rhythmic alignment, underscoring the importance of selecting compatible songs and transition segments. The system enhances both tonal and rhythmic continuity by leveraging MERT embeddings for transition segment selection and employing MusicGen for transition generation, resulting in a more natural listening experience. The results highlight the significance of structured playlist ordering, optimal transition segment detection, and music generation in improving transition quality. As AI driven music tools continue to evolve, frameworks like DJ AI have the potential to redefine how music is mixed and enjoyed, making seamless transitions accessible not only to professional DJs but also to casual listeners seeking an enhanced auditory experience.

REFERENCES

[1] Rachel M. Bittner, Minwei Gu, Gandalf Hernandez, Eric J., Humphrey, Tristan Jehan, P. Hunter McCurry, and Nicola Montecchio. 2016. AUTOMATIC PLAYLIST SEQUENCING AND TRANSITIONS. (2016). https://rachelbittner.weebly.com/uploads/3/2/1/8/32182799/bittner_ismir-playlist_2017.pdf

[2] J. Copet, F. Kreuk, I. Gat, et al. 2024. Simple and Controllable Music Generation. In *NeurIPS*. https://github.com/facebookresearch/audiocraft

[3] D. P. W. Ellis and G. E. Poliner. 2007. Identifying 'Cover Songs' with Chroma Features and Dynamic Programming Beat Tracking. In *IEEE ICASSP*. 1429–1432. https://ieeexplore.ieee.org/document/4218379

[4] S. Gossett. 2006. The Camelot Sound: A Guide to Harmonic Mixing. *DJ Techniques Magazine* (2006).

[5] Jia-Lien Hsu and Shuh-Jiun Chang. 2021. Generating Music Transition by Using a Transformer-Based Model. *Electronics* 10, 18 (2021). https://doi.org/10.3390/electronics10182276

[6] S. Jun, D. Kim, and Y. Kim. 2013. Social Mix: Automatic Music Recommendation and Mixing System Based on Social Network Analysis. *Expert Systems with Applications* 40, 7 (2013), 2601–2610. https://doi.org/10.1007/s11227-014-1182-1

[7] K. Kilgour, M. Zuluaga, D. Roblek, and M. Sharifi. 2018. Fréchet Audio Distance: A Reference-Free Metric for Evaluating Music Enhancement Algorithms. In *Interspeech*. 106–110. https://www.isca-archive.org/interspeech_2019/kilgour19_interspeech.pdf

[8] C. L. Krumhansl and M. A. Schmuckler. 1990. Key-Finding Algorithms and the Perception of Tonal Structure. *Psychological Review* 97, 2 (1990), 211–222. https://doi.org/10.1037/0096-1523.31.5.1124

[9] R. F. Kubichek. 1993. Mel-cepstral Distance Measure for Objective Speech Quality Assessment. In *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*. 125–128. https://ieeexplore.ieee.org/document/407206

[10] Y. Li, R. Yuan, G. Zhang, et al. 2024. MERT: Acoustic Music Understanding Model with Large-Scale Self-Supervised Training. In *ICLR*. https://github.com/yizhilll/MERT

[11] M. Müller. 2007. Dynamic Time Warping. In *Information Retrieval for Music and Motion*. Springer, 69–84.

[12] Kyle Robinson and Dan Brown. 2023. Automated Time-frequency Domain Audio Crossfades using Graph Cuts. arXiv:2301.13380 [cs.SD] https://arxiv.org/abs/2301.13380

[13] P. Senin. 2008. *Dynamic Time Warping Algorithm Review*. Technical Report. University of Hawaii at Manoa.

[14] L. Wang, Q. Zhao, X. Wang, et al. 2021. Embedding Normalization: Significance Preserving Feature Normalization for Click-Through Rate Prediction. *IEEE Transactions on Knowledge and Data Engineering* (2021). https://ieeexplore.ieee.org/document/9679881

[15] M. Yu, M. R. Gormley, and M. Dredze. 2015. Combining Word Embeddings and Feature Embeddings for Fine-grained Relation Extraction. In *NAACL-HLT*. 1374–1379.