On the Distillation of Stories for Transferring Narrative Arcs in Collections of Independent Media

Dylan R. Ashley [1] [*]

Vincent Herrmann [1] [*]

Zachary Friggstad [2]

Jürgen Schmidhuber [1,3]

[1] The Swiss AI Lab IDSIA (USI/SUPSI), Lugano, Switzerland
[2] University of Alberta, Edmonton, Canada
[3] AI Initiative, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia

Abstract

The act of telling stories is a fundamental part of what it means to be human. This work introduces the concept of narrative information, which we define to be the overlap in information space between a story and the items that compose the story. Using contrastive learning methods, we show how modern artificial neural networks can be leveraged to distill stories and extract a representation of the narrative information. We then demonstrate how evolutionary algorithms can leverage this to extract a set of narrative templates and how these templates—in tandem with a novel curve-fitting algorithm we introduce—can reorder music albums to automatically induce stories in them. In the process of doing so, we give strong statistical evidence that these narrative information templates are present in existing albums. While we experiment only with music albums here, the premises of our work extend to any form of (largely) independent media.

1 Introduction

When presenting a media collection, the sequence in which items are displayed can significantly impact the overall narrative and impression. While this ordering is often meticulously curated in venues like art galleries, larger collections or less prioritized venues might rely on arbitrary arrangements for cost efficiency. We introduce a data-driven method to convey desired narratives by organizing media. Specifically, we use information theory and deep learning to distill elements of stories down to an essential form. We then use evolutionary algorithms and a novel curve-fitting algorithm to learn a set of template curves and fit collections to these templates.

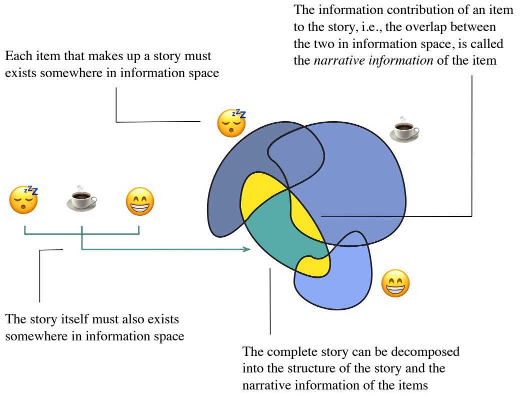

The information contribution of an item to the story, i.e., the overlap between the two in information space, is called the *narrative information* of the item
Each item that makes up a story must exists somewhere in information space
The story itself must also exists somewhere in information space
The complete story can be decomposed into the structure of the story and the narrative information of the items

Figure 1: If a story is a structured collection of items, then—in information space—the story must have some overlap with each of the items. We call this overlap the *narrative information* of the items.

[*] Equal contribution. Correspondence to dylan.ashley@idsia.ch or vincent.herrmann@idsia.ch

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

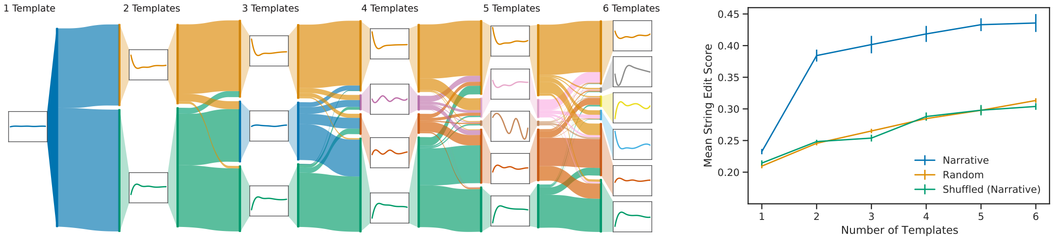


Figure 2: **(left)** Assignment of individual songs to increasing numbers of template curves learned with narrative essence. Colors show post-hoc analysis to try and find similar prototypical curves. **(right)** The performance on the FMA validation set of the templates learned with narrative essence.

2 Narrative Essence and Story Templates

We study stories as collections of basic elements (atoms) and their meaningful ordering, termed the narrative. The *narrative information* of an atom is its relevance to the overall story, depicted in Figure 1. We propose that ordering a media collection can create a desired narrative, allowing for different storytelling structures like climaxes or gradual build-ups. Every atom has inherent properties influencing its role in a story. This leads to the concept of *narrative essence*, a learned representation emphasizing an atom's most narrative-relevant features. Formally, we define as narrative essence $f_E(x)$ of atom x, generated by a feature extractor f_E, as the feature which maximizes the mutual information between the unordered set of features of the atoms in a collection c and the ground truth order $o(c)$ of c: $f_E = \arg\max_f I\big(\{f(x)|x \in c\}; o(c)\big)$. As described in Appendix C, we use contrastive learning to learn a feature extractor f_E for music albums from the FMA dataset [4].

Narrative arcs, present in dramatic arts like novels and plays, vary in structure (e.g., tragedies, comedies). Different media have distinct arcs. Using the genetic algorithms described in Appendix D and our learned narrative essence extractor, we derive story template curves from music albums in the FMA dataset. Figure 2, left, shows the narrative templates found for the training split. We employ a novel curve-fitting algorithm (described in Appendix E) to assess the templates' accuracy and compare the original album ordering to our fitted sequences using a string edit score. Our results, compared against both random orderings and shuffled in-album narrative essence scores (see Figure 2, right), show that the narrative essence and the learned templates at least partially explain existing album ordering ($p < 0.05$; see Appendix B).

3 Related Work

The use of machine learning to derive narrative arcs has previously been explored by Reagan et al. [15]—who derive the emotional arc of stories from a corpus of English texts. Mathewson et al. [14] use an information-theoretic approach to design a narrative arc and applied this to dialogue generation. There exists a considerable body of work in music playlist continuation [2, 13, 19]. For music playlist ordering, work remains sparse. For a detailed overview of spatial representation of musical form, see Bonds [1]. Contrastive methods for learning representations of perceptual data [3, 6], including music [17], have recently gained much attention. Most similar to our approach of learning narrative essence are methods that maximize mutual information between local and global representations [8, 12].

4 Conclusion

We explored how stories manifest in independent media collections, introducing the concept of narrative essence to capture an element's role in the narrative. Using neural networks and contrastive learning, we learned a feature extractor that distills music tracks to their narrative essence. We then used genetic algorithms to learn a set of narrative template curves and a new curve-fitting algorithm to order collections using these templates. We gave statistical evidence that the ordering of music albums can be partially explained by our system. Though our focus was on music, our approach applies to any (largely) independent set of media. We give a demonstration of the practical outcome of this work in Appendix A.

5 Ethical Implications

The above work seeks to automate part of the artistic process and so may empower artists, but could likewise reduce their demand and potentially put individuals' immediate employment at risk. Beyond this, the authors foresee no notable ethical implications of this work. The authors would like to encourage anyone noting such implications to reach out to the authors directly.

Acknowledgements

We want to thank Kory Mathewson and DeepMind Technologies Limited for comments on an earlier version of this work. This work was supported by the European Research Council (ERC, Advanced Grant Number 742870) and the Swiss National Supercomputing Centre (CSCS, Project s1090).

References

Bonds, M. E. (2010). The spatial representation of musical form. *Journal of Musicology*, *27*(3), 265–303. https://doi.org/10.1525/jm.2010.27.3.265

Bonnin, G., & Jannach, D. (2013). A comparison of playlist generation strategies for music recommendation and a new baseline scheme. *Papers from the 2013 AAAI Workshop*, 16–23. https://www.aaai.org/ocs/index.php/WS/AAAIW13/paper/view/7078

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. *International conference on machine learning*, 1597–1607.

Defferrard, M., Benzi, K., Vandergheynst, P., & Bresson, X. (2017). FMA: A dataset for music analysis, 316–323. https://ismir2017.smcnus.org/wp-content/uploads/2017/10/75%5C_Paper.pdf

Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional lstm networks. *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, *4*, 2047–2052.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al. (2020). Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, *33*, 21271–21284.

Gutmann, M., & Hyvärinen, A. (2010). Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 297–304.

Hjelm, R. D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Trischler, A., & Bengio, Y. (2018). *Learning deep representations by mutual information estimation and maximization*. arXiv. http://arxiv.org/abs/1808.06670

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hopcroft, J. E., & Karp, R. M. (1973). An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing*, *2*(4), 225–231. https://doi.org/10.1137/0202019

Jonker, R., & Volgenant, A. (1987). A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, *38*(4), 325–340. https://doi.org/10.1007/BF02278710

Löwe, S., O'Connor, P., & Veeling, B. (2019). Putting an end to end-to-end: Gradient-isolated learning of representations. *Advances in neural information processing systems*, *32*.

Maillet, F., Eck, D., Desjardins, G., & Lamere, P. (2009). Steerable playlist generation by learning song similarity from radio station playlists. *Proceedings of the 10th International Society for Music Information Retrieval Conference*, 345–350. https://archives.ismir.net/ismir2009/paper/000012.pdf

Mathewson, K. W., Castro, P. S., Cherry, C., Foster, G. F., & Bellemare, M. G. (2020). Shaping the narrative arc: Information-theoretic collaborative dialogue. *Proceedings of the 11th International Conference on Computational Creativity*, 9–16. http://computationalcreativity.net/iccc20/papers/010-iccc20.pdf

Reagan, A. J., Mitchell, L., Kiley, D., Danforth, C. M., & Dodds, P. S. (2016). The emotional arcs of stories are dominated by six basic shapes. *EPJ Data Science*, *5*(1), 31–42. https://doi.org/10.1140/epjds/s13688-016-0093-1

RIAA. (2021). *Gold & platinum*. Recording Industry Association of America. https://www.riaa.com/gold-platinum/?tab_active=awards_by_album

Saeed, A., Grangier, D., & Zeghidour, N. (2021). Contrastive learning of general-purpose audio representations. *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3875–3879.

Schubert, E. (1999). Measuring emotion continuously: Validity and reliability of the two-dimensional emotion-space. *Australian Journal of Psychology*, *51*(3), 154–165.

Vall, A., Quadrana, M., Schedl, M., & Widmer, G. (2019). Order, context and popularity bias in next-song recommendations. *International Journal of Multimedia Information Retrieval*, *8*(2), 101–113. https://doi.org/10.1007/s13735-019-00169-8

van den Oord, A., Li, Y., & Vinyals, O. (2018). *Representation learning with contrastive predictive coding*. arXiv. http://arxiv.org/abs/1807.03748

A Demonstration

Figure 3 in gives a demonstration of the practical application of this work. Here, we used the narrative essence of the tracks in Michael Jackson's *Thriller* to fit the album to the four distinct narrative template curves given in Figure 2, left. By doing so, we have induced several different stories in the album. The methods presented here can trivially carry out the same task with any album.

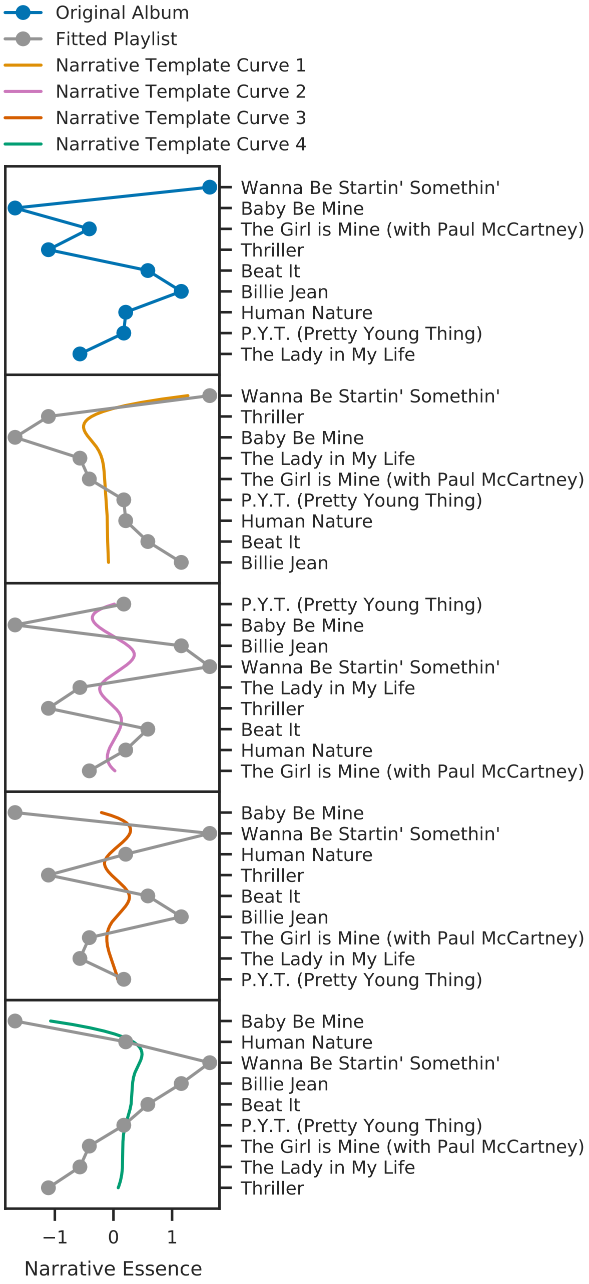


Figure 3: Narrative essence of the album *Thriller* by Michael Jackson—the best-selling original album of all time [16]—in the original order, and fitted to the four narrative template curves found using the method described in Section 2.

B Evidence for the Existence of Story Templates

An important question we must address while looking at Figure 2, right, is whether or not the improvement of the learned curves over the baselines is significant. This is equivalent to asking if the order of the albums is partially explained by the valence and thus if the narrative structures discovered by our algorithm exist within music albums. To answer this, we compare the mean string edit score for the selected $k = 4$ templates with the mean string edit score for both baselines on the test set. We find that the difference observed is significant with a family-wise error rate of $p < 0.05$ using t-tests with Holm-Bonferroni corrections.

C Learning Narrative Essence From Data

A narrative essence extractor f_E can be learned using a dataset of ordered media collections. We do this using noise contrastive estimation [7]—specifically, a modification of InfoNCE [20]. The idea, as shown in Figure 4, is the following: we give each item in a collection to a learnable feature extractor f_θ: a neural network with parameters θ. A second learnable function g_ϕ: a recurrent neural network with parameters ϕ, takes a sequence s of features as input and produces a scalar score $g_\phi(s)$. If g_ϕ receives a sequence in the correct ground-truth order, $s^* = (f_\theta(x_1), f_\theta(x_2), f_\theta(x_3), ...)$, it should produce a high score. For randomly ordered sequences, it should produce a low score. This can only be achieved by g_ϕ if (1) the correct orders of the collections in our dataset have some property that distinguishes them from random orders, and (2) f_θ learns some atom-wise feature that lets g_ϕ recognize this property. Feature extractor f_θ and sequence model g_ϕ are jointly trained to minimize the loss

$$\mathcal{L}_{\mathrm{N}}(\theta, \phi; \mathcal{D}) = -\mathbb{E}_{S\sim\mathcal{D}}\left[\log \frac{g_\phi(s^*)}{\sum_{s\in S} g_\phi(s)}\right], \tag{1}$$

where $\mathcal{D}$ is the dataset of collections with a ground truth order, and S is a set of N sequences that include the correctly ordered sequence s^*. The other $N-1$ sequences in S are random permutations of s^*. The extracted features should be normalized across the sequence so that g_ϕ considers the relative, and not the absolute value, of the extracted feature.

In analogy to van den Oord et al. [20], we prove in Appendix C.5 that minimizing $\mathcal{L}_{\mathrm{N}}$ maximizes a lower bound on the narrative information, i.e., the mutual information between the atom-wise features extracted by f_θ and the order of the collection:

$$I\big(\{f_\theta(x)|x \in c\}; o(c)\big) \geq \log(N) - \mathcal{L}_{\mathrm{N}}. \tag{2}$$

C.1 Narrative Essence in Music Albums

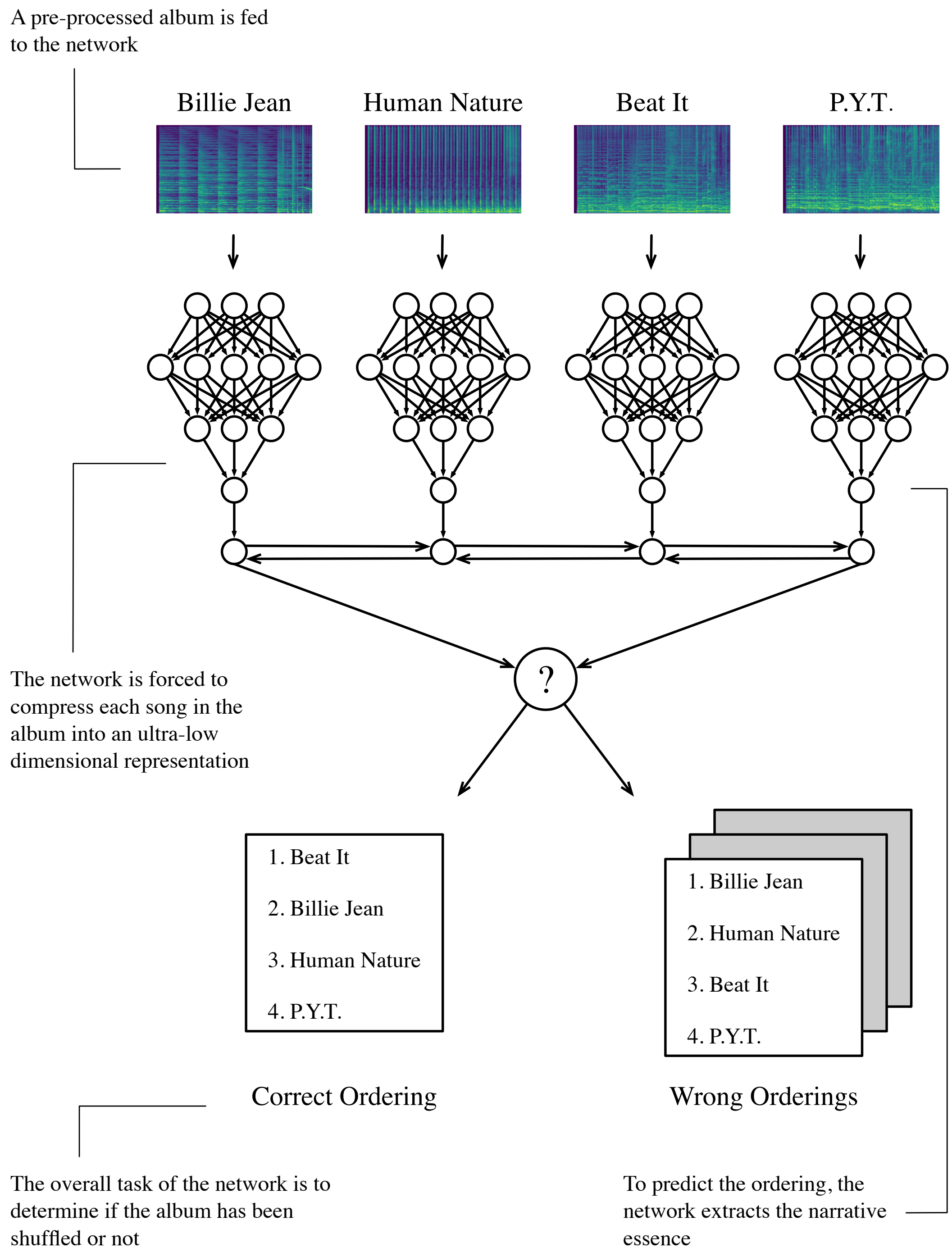
We empirically investigate the concept of narrative essence using the example of music albums. We selected the FMA dataset [4] as it is—at the time of writing—the largest open music album dataset that includes raw audio files.

While, in principle, highly sophisticated and specialized feature extraction architectures could be used for f_θ, in our experiments, we restrict ourselves to relatively simple and computationally cheap methods. Each track is represented by features commonly used in music information retrieval that come pre-computed with the available dataset. These features form a sequence of 75 vectors of size 7 (for more details, see Section C.3). This sequence is the input to the feature extractor f_θ, for which we use a bidirectional LSTM model [5, 9]. We choose a recurrent feature encoder instead of a feed-forward architecture to give f_θ more powerful conditional processing abilities.

The output of f_θ is the narrative essence of the given song. In principle, the narrative essence could be a vector of any size. However, Table 1, which shows the results of twenty-five runs using the FMA validation set, demonstrates that a higher dimensional narrative essence leads to only marginal improvements in mutual information captured. These diminishing returns provide strong evidence that narrative essence, at least for songs in the context of a music album, can be represented as a scalar value. Note that even a low-dimensional version of narrative essence still captures something much more sophisticated than a basic ranking. A benefit of using a scalar for narrative essence is that it is directly comparable to other available scalar features (see Section C.2).

Like in Figure 4, we model g_ϕ as a bidirectional LSTM as well. It takes a sequence of narrative essence features as input and computes a scalar score. In comparison to f_θ, g_ϕ has a lower capacity (fewer learnable parameters and more regularization) because there are much fewer full collections (albums) than individual items (songs). Thus g_ϕ is at a considerable risk of overfitting. The specific hyperparameters we use are provided in Appendix C.4. When trained on the full FMA training set[2], the extracted narrative essence achieves a mutual information with the album ordering, as determined by Equation 2, of ca. 1.924 bits on the validation set.

[2] Note that here and everywhere else we have excluded albums with less than 3 or more than 20 tracks.



A pre-processed album is fed to the network

The network is forced to compress each song in the album into an ultra-low dimensional representation

The overall task of the network is to determine if the album has been shuffled or not

To predict the ordering, the network extracts the narrative essence

Figure 4: To learn a network that can distill music tracks down to their narrative essence, we feed a bidirectional recurrent neural network a pre-processed album and train it to determine whether the album its been given has been shuffled or not. By creating a bottleneck between the encoder and the recurrent layer we can control the dimensionality of the learned narrative essence.

Table 1: Representation size and captured narrative information

Narrative Essence Size	**Mutual Information (in bits)**
1	1.924 ± 0.0296
2	1.936 ± 0.0183
4	1.957 ± 0.0217
8	1.950 ± 0.0216
16	1.975 ± 0.0150

C.2 Narrative Essence in Comparison With Other Features

When treating the feature extractor f_θ as fixed and only learning the scoring model g_ϕ to minimize $\mathcal{L}_\mathrm{N}$, we can use Equation 2 to approximate the mutual information between any available feature and the collection orders. Here, we compare the narrative essence feature extracted using f_θ learned on the FMA dataset with some of the other features available in the dataset. To do so, we learn a dedicated scoring model g_ϕ for each available feature—including energy, tempo and valence: a feature designed to capture the mood of a song, roughly ranging from sad to happy [18]. As shown in Figure 5, narrative essence has more mutual information with the album order than any of the other features. The mutual information lower bounds seen in Figure 5 are significantly lower than the ones achieved when training on the full dataset (compare Table 1). This discrepancy is because only a limited subset of the FMA dataset includes the listed pre-computed features when learning the scoring models. Nevertheless, these results show that our formulation of narrative essence, in combination with the very general processing techniques (i.e., global track feature statistics and an LSTM encoder), robustly outperforms highly engineered features as a candidate for the narrative.

Figure 6 shows the correlation of the learned narrative essence feature with eight other features. Here we see a high correlation with the features we expect to be associated (i.e., valence, energy) and a low correlation with more technical or applied features like acousticness, instrumentalness, and liveness. Note that the orientation (sign) of the narrative essence is simply a random product of the initialization and has no further meaning; the negative of the narrative essence would have exactly the same amount of mutual information with the collection order.

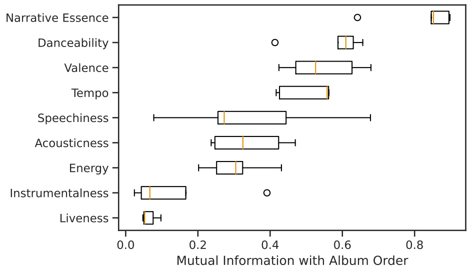


Figure 5: The lower bound of the mutual information in bits between different features and the album order, calculated on the subset of the FMA validation set that includes valence. Results are shown over five seeds.

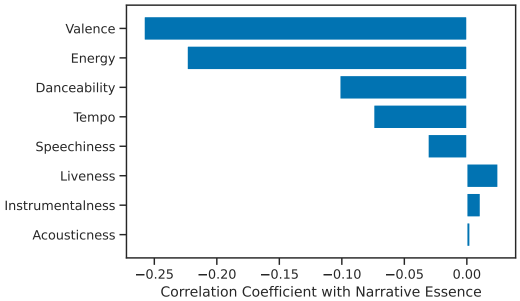


Figure 6: The Pearson correlation coefficient of different features with the narrative essence on the subset of the FMA dataset that includes valence.

C.3 Track Input Features

The FMA dataset provides the following track features: 12 Chroma features, 6 Tonnetz features, 20 MFCC features, Spectral centroid, Spectral bandwidth, 7 Spectral contrast features, Spectral rolloff, RMS energy and Zero-crossing rate.

For every feature, 7 global statistical properties are given: mean, standard deviation, skew, kurtosis, median, minimum and maximum. We treat these statistical properties as a vector and construct a sequence of these vectors from the 75 features, which constitutes the input for the narrative essence extractor f_θ. The length of this feature sequence is constant, independent of the track's length.

For tracks that are not included in the FMA data, these features can easily be computed directly from audio standard MIR techniques (the implementation is provided by [4]). Before giving the sequence to g_ϕ, learnable start- and end-of-sequence tokens are added.

C.4 Model Hyperparameters

All models described in this section have the same hyperparameters. The batch size is 16, N is 32 (that means we have 31 negative samples for each example in the batch).

The feature encoder f_θ is a bidirectional LSTM with 2 layers, 7 input features, 128 hidden units and a sigmoid output nonlinearity. For regularization we use dropout of 0.1 and no weight decay.

The sequence scoring model g_ϕ is also bidirectional LSTM with 2 layers. It has 32 hidden units and no output nonlinearity. For regularization we use no dropout and a weight decay of 10^{-5}.

For both models, we use the Adam optimizer with a learning rate of 10^{-4}, and early stopping based on the validation loss.

C.5 Narrative Essence and Mutual Information

Recall that c is an unordered collection of items x, and $o(c)$ is its correct order. S is a set of N sequences of the encoded items, containing the correct sequence $s^* = (f_\theta(x_1), f_\theta(x_2), f_\theta(x_3), ...)$ (i.e., the one adhering to $o(c)$), and $N-1$ random permutations of s^*. The probability that a particular sequence s_i from S is the correct sequence s^* is

$$
\begin{aligned}
p(s_i = s^* | S, o(c)) &= \frac{p(s_i = s^*, S | o(c))}{\sum_j p(s_j = s^*, S | o(c))} \\
&= \frac{p(s_i = s^*) p(S | s_i = s^*, o(c))}{\sum_j p(s_j = s^*) p(S | s_j = s^*, o(c))} \\
&= \frac{p(s_i = s^*) p(s_i | o(c)) \prod_{l \neq i} p(s_l)}{\sum_j p(s_j = s^*) p(s_j | o(c)) \prod_{l \neq j} p(s_l))} \\
&= \frac{\frac{1}{N} p(s_i | o(c)) \prod_{l \neq i} p(s_l)}{\sum_j \frac{1}{N} p(s_j | o(c)) \prod_{l \neq j} p(s_l))} \\
&= \frac{p(s_i | o(c)) \prod_{l \neq i} p(s_l)}{\sum_j p(s_j | o(c)) \prod_{l \neq j} p(s_l))} \cdot \frac{\prod_k p(s_k)}{\prod_k p(s_k)} \\
&= \frac{\frac{p(s_i | o(c))}{p(s_i)}}{\sum_j \frac{p(s_j | o(c))}{p(s_j)}}.
\end{aligned}
$$

With Equation 1, $g_\phi(s)$ is trained to estimate the density ratio $\frac{p(s|o(c))}{p(s)}$. This means that we can write (following the steps from [20])

$$
\begin{aligned}
\mathcal{L}_{\mathrm{N}}^{\mathrm{opt}} &= -\mathbb{E}_{S \sim \mathcal{D}} \log \left[\frac{\frac{p(s^* | o(c))}{p(s^*)}}{\frac{p(s^* | o(c))}{p(s^*)} + \sum_{s \in S_{\mathrm{neg}}} \frac{p(s | o(c))}{p(s)}} \right] \\
&\approx \mathbb{E}_{S \sim \mathcal{D}} \log \left[1 + \frac{p(s^*)}{p(s^* | o(c))} (N-1) \right] \\
&\geq \mathbb{E}_{S \sim \mathcal{D}} \log \left[\frac{p(s^*, o(c))}{p(s^*) p(o(c))} N \right] \\
&= -I(s^*; o(c)) + \log(N) \\
&= -I(f_E(x_1), f_E(x_2), f_E(x_3), ...; o(c)) + \log(N).
\end{aligned}
$$

D Story Template Extraction Algorithm

Extracting a set of narrative arc templates from a collection of albums can be done using Algorithm 1. Note that this algorithm is general and can make use of any collection of media if the narrative essence is replaced by a semantically similar metric. In our experiments, we always used $\mathbf{x} = [0.0, 0.2, 0.3, 0.5, 0.65, 0.8, 1.0]^T$. To derive the value of a template at a given x, cubic-spline interpolation is recommended; for the cost of fitting an album to a template, using the mean-squared error is recommended.

Algorithm 1 Story Template Extraction

Input: $\mathbf{x} = [x_0, x_1, ..., x_q]^T$ where x_i is the relative position of the i-th point in the desired templates and a set of albums $\{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_n\}$ with each $\mathbf{a}_i = \{(u_0, v_0), (u_1, v_1), ..., (u_m, v_m)\}^T$ where u_j is the relative position of track j in the album, and v_j is the normalized narrative essence of track j

Output: set of templates $\{\mathbf{t}_1, \mathbf{t}_2, ..., \mathbf{t}_p\}$ with each $\mathbf{t}_i = [y, y_1, ..., y_q]^T$ where y_j is the normalized narrative essence of the j-th point in the template

```
$s \leftarrow$ population size
$b \leftarrow$ number of children for each generation
for $i \in \{1..s\}$ do
    for $j \in \{1..p\}$ do
        for $k \in \{1..q\}$ do
            $P[i, j, k] \leftarrow \mathcal{N}(0, 1)$
        end for
    end for
end for

while not done do
    $\sigma = \mathcal{N}(0, 1)$
    for $i \in \{1..b\}$ do
        $father \leftarrow$ random integer in $\{0, 1, ..., s\}$
        $mother \leftarrow$ random integer in $\{0, 1, ..., s\} - \{father\}$
        for $j \in \{1..p\}$ do
            for $k \in \{1..q\}$ do
                $P[s+i, j, k] \leftarrow P[father, j, k]$ with probability $p$ and $P[mother, j, k]$ with probability $1 - p$
                $P[s+i, j, k] \leftarrow P[s+i, j, k] + \mathcal{N}(0, \sigma)$
            end for
        end for
    end for
    for $i \in \{1..(b+s)\}$ do
        $\mathbf{c}_i \leftarrow$ minimum cost as defined in Equation 3 for fitting albums using the templates $P[i, :, :]$
    end for
    order $P$ in increasing order of corresponding $\mathbf{c}$
    $P \leftarrow P[1 : s, :, :]$
end while
return $P$
```

While many different cost functions for a set of templates could be used here, we use the following:

$$\mathbf{c} = \sum_{i=1}^{n} \min_{p} \frac{1}{l_i} \sum_{j=1}^{l_i} \left(v_i(j) - t_p(j_r)\right)^2, \tag{3}$$

where n is the number of albums in the training set, l_i the number of tracks in album i, $v_i(j)$ the normalized narrative essence value of the jth track of album i, and $t_p(j_r)$ is the value of template p at the relative position $j_r = (j-1)/(l_i-1)$. We learn these templates using the training split provided by the FMA dataset and evaluate them on the validation split by fitting the narrative essence of each album to the templates using the algorithm given in Appendix E.

E Template Curve Fitting Algorithm

Deriving an ordering of the media such that their respective values fit a narrative template can be done using Algorithm 2. The ordering Algorithm 2 finds will be minimal first in the maximum deviation of a value from the template curve and minimal second in the average deviation of values from the template curve. For n items, the worst-case time complexity of this algorithm—provided efficient bipartite matching algorithms such as Hopcroft-Karp [10] and LAPJVsp [11] are used—is in $O(n^3)$. In most applications of this work—and for all but the largest collections of independent media—extracting the values that will be fitted will consume vastly more time than the fitting itself.

Algorithm 2 Template Curve Fitting

```
Input: normalized values to fit $\mathbf{y}$ and template curve function $f$ with domain and range $[0, 1]$
Output: ordering $\mathbf{x}$ over values $\mathbf{y}$ such that the $i$-th value in the ordering is $\mathbf{x}[i]$
$\mathbf{z} \leftarrow \left[f(\frac{0}{|\mathbf{y}|-1}), f(\frac{1}{|\mathbf{y}|-1}), ..., f(\frac{|\mathbf{y}|-1}{|\mathbf{y}|-1})\right]^T$
$\mathbf{d} \leftarrow \mathbf{y}\mathbf{z}^T$

$a \leftarrow 1$
$b \leftarrow |\mathbf{d}|$
while $a \neq b$ do
    $p \leftarrow a + \lfloor (b-a)/2 \rfloor$
    $L, R \leftarrow \{1..|\mathbf{y}|\}$
    $E \leftarrow \{(i \in L, j \in R) \mid \|\mathbf{y}[i] - \mathbf{z}[j]\| \leq \mathbf{d}[p]\}$
    if $\exists$ perfect matching for bipartite graph $(L, R, E)$ then
        $b \leftarrow p$
    else
        $a \leftarrow p + 1$
    end if
end while

$L, R \leftarrow \{1..|\mathbf{y}|\}$
$E \leftarrow \{(i \in L, j \in R, \|\mathbf{y}[i] - \mathbf{z}[j]\|) \mid \|\mathbf{y}[i] - \mathbf{z}[j]\| \leq \mathbf{d}[a]\}$
$M \leftarrow$ minimum-cost perfect matching for weighted bipartite graph $(L, R, E)$
for $i \in \{1..|\mathbf{y}|\}$ do
    for $j \in \{1..|\mathbf{y}|\}$ do
        if $(i, j) \in M$ then
            $\mathbf{x}[j] = i$
        end if
    end for
end for

return $\mathbf{x}$
```

F Source Code

A practical implementation of this paper is available at https://github.com/dylanashley/story-distiller

An implementation of the algorithm described in Appendix C can be found at https://github.com/vincentherrmann/narrative-essence

An implementation of the algorithm described in Appendix D can be found at https://github.com/dylanashley/story-template-extraction

An implementation of the algorithm described in Appendix E can be found at https://github.com/dylanashley/playlist-story-builder