

An Inverse Partial Optimal Transport Framework for Music-guided Movie Trailer Generation

Yutong Wang*

Beijing Institute of Technology, Beijing, China
yutong.wang@bit.edu.cn

Hongteng Xu

Renmin University of China, Beijing, China
hongtengxu@ruc.edu.cn

Sidan Zhu*

Beijing Institute of Technology, Beijing, China
sidan_zhu@bit.edu.cn

Dixin Luo[†]

Beijing Institute of Technology, Beijing, China
Key Laboratory of Artificial Intelligence,
Ministry of Education, Shanghai, China
dixin.luo@bit.edu.cn

Abstract

Trailer generation is a challenging video clipping task that aims to select highlighting shots from long videos like movies and reorganize them in an attractive way. In this study, we propose an inverse partial optimal transport (IPOT) framework to achieve music-guided movie trailer generation. In particular, we formulate the trailer generation task as selecting and sorting key movie shots based on audio shots, which involves matching the latent representations across visual and acoustic modalities. We learn a multi-modal latent representation model in the proposed IPOT framework to achieve this aim. In this framework, a two-tower encoder derives the latent representations of movie and music shots, respectively, and an attention-assisted Sinkhorn matching network parameterizes the grounding distance between the shots' latent representations and the distribution of the movie shots. Taking the correspondence between the movie shots and its trailer music shots as the observed optimal transport plan defined on the grounding distances, we learn the model by solving an inverse partial optimal transport problem, leading to a bi-level optimization strategy. We collect real-world movies and their trailers to construct a dataset with abundant label information called CMTD and, accordingly, train and evaluate various automatic trailer generators. Compared with state-of-the-art methods, our IPOT method consistently shows superiority in subjective visual effects and objective quantitative measurements. The code is available at <https://github.com/Dixin-Lab/Automatic-Movie-Trailer-Generator>.

CCS Concepts

• **Computing methodologies** → **Matching; Learning latent representations; Video summarization.**

*Both authors contributed equally to this work.

[†]Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MM '24, October 28–November 1, 2024, Melbourne, VIC, Australia.

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 979-8-4007-0686-8/24/10

<https://doi.org/10.1145/XXXXXX.XXXXXX>

Keywords

Trailer generation, video clipping, inverse optimal transport, movie-trailer dataset

ACM Reference Format:

Yutong Wang, Sidan Zhu, Hongteng Xu, and Dixin Luo. 2024. An Inverse Partial Optimal Transport Framework for Music-guided Movie Trailer Generation. In *Proceedings of the 32nd ACM International Conference on Multimedia (MM '24)*, October 28–November 1, 2024, Melbourne, VIC, Australia. ACM, New York, NY, USA, 13 pages. <https://doi.org/10.1145/XXXXXX.XXXXXX>

1 Introduction

As collections of movie highlights that may attract audiences, trailers play a central role in movie promotion. Unlike video summarization [32, 33, 51], which selects key frames or shots to alleviate the redundancy of video content while keeping the completeness of the storyline, trailer generation [20, 34, 41] needs to select attractive movie highlights but reorganize them to hide the original movie's storyline to some extent. The selection and reorganization of movie shots are determined by various factors, e.g., the semantics of background music, the synchronization of visual and acoustic content, the logical flow of characters' dialogues, and so on, which requires a deep understanding of the movie. Therefore, generating a high-quality movie trailer involves sophisticated video clipping and editing, which is time-consuming and labor-intensive (and thus, expensive). Typically, the trailer of a Hollywood blockbuster may require months of work by a team of professional editors to select the movie highlights and align them with background music.

Due to the above fact, many academic and industrial researchers have made efforts to improve the efficiency of movie trailer generation, gradually making the whole process automatic. Currently, some music-guided movie trailer generation methods, especially those learning-based ones [26, 43, 57], have been proposed, which generate trailers from movies automatically based on given background music. At the same time, some commercial software like Muvee [12] is developed to achieve music-guided video clipping and trailer generation. However, when utilizing background music, these methods mainly focus on synchronizing movie shots according to the music rhythm while ignoring the semantic alignment between visual and acoustic information. As a result, the performance of the methods is still unsatisfactory in practical applications. What is worse, the learning-based methods often suffer from the

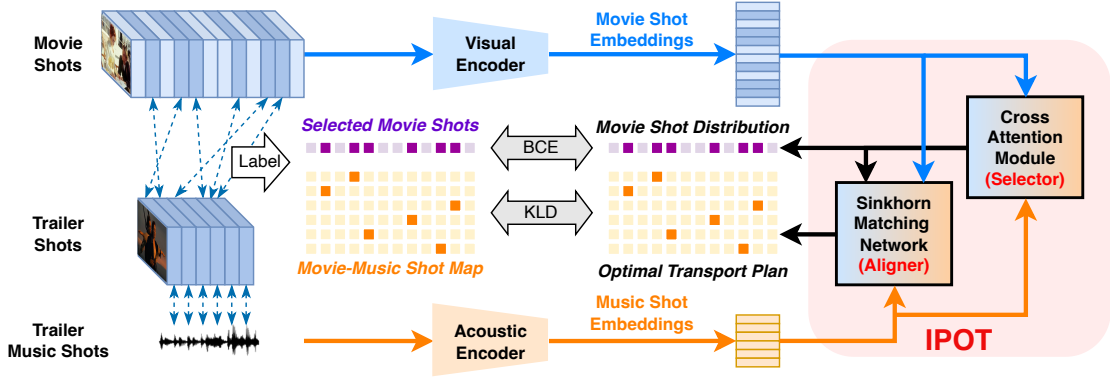


Figure 1: An illustration of our IPOT framework for learning a music-guided movie trailer generator.

scarcity of labeled training data. For example, the point process-based method in [57] needs to learn an attractiveness model based on the movies with audiences' fixation information collected by professional eye trackers. The emotion correlation-based method in [26] requires video and audio shots to be labeled with manually defined emotion categories. Because such annotation is difficult and time-consuming, the datasets they used are limited in size, leading to a high risk of overfitting.

To overcome the above challenges and boost the performance of automatic trailer generation, in this study, we propose a novel music-guided movie trailer generation method with the help of computational optimal transport techniques. As illustrated in Figure 1, we formulate the music-guided trailer generation task as selecting and sorting key movie shots based on given audio shots and establish an inverse partial optimal transport (IPOT) framework to learn a model to achieve this aim. In particular, given a movie and its corresponding trailer, we first leverage a two-tower encoder to obtain the latent representations of movie shots and trailer music shots, respectively. Given the visual and acoustic latent representations, our IPOT framework applies an attention-assisted Sinkhorn matching network to parameterize the distribution of movie shots and the grounding distances between the latent representation across the two modalities. Accordingly, whether a movie shot should be selected to construct a trailer or not is determined by the movie shot distribution, and the alignment between the movie and music shots is achieved by solving a partial optimal transport problem based on the grounding distances and the movie shot distribution. In the training phase, we learn the model by fitting the movie shot distribution and the cross-modal optimal transport plan to the ground truth movie shot indicator and movie-music shot map, leading to the proposed IPOT framework. A bi-level optimization strategy is applied to learn the model effectively, leading to a movie shot selector and a movie-music shot aligner.

To train our model and compare it with state-of-the-art trailer generators, we collect real-world movies and their official trailers and construct a comprehensive movie-trailer dataset (CMTD) with abundant label information. Compared to existing movie-trailer datasets [11, 19, 49], CMTD contains multiple official trailers for each movie and provides segmentation information for shots in all movies, trailers, and corresponding trailer music. The movie shots

and music shots are aligned by matching the movie shots with the corresponding trailer shots. In order to adapt to different tasks and future studies, we also provide the metadata related to the movies, such as subtitles, synopsis, turning points annotations, and so on. To the best of our knowledge, CMTD might be the largest labeled movie-trailer dataset at the current stage.

In summary, the contributions of this work include two folds:

- We propose a novel and effective IPOT framework for music-guided movie trailer generation. Applying the proposed learning framework leads to a new optimal transport-based solution to music-guided movie trailer generation task.
- We construct a new public comprehensive movie-trailer dataset for movie trailer generation and future video understanding tasks. We train and evaluate various trailer generators on the dataset. Experimental results demonstrate the superiority of our IPOT-based trailer generator on both objective measurements and subjective effects.

2 Related Work

2.1 Video understanding and trailer generation

Video understanding is an extensive research field that involves exploring the semantics of video content and aligning it with other modalities, such as texts and audio. As typical video understanding tasks, video retrieval [7, 10] aims to search videos according to their relevance to textual queries, and video tagging [30, 50] and temporal action localization [27, 60, 62] aim to annotate videos or the scenes in them automatically. The development of deep learning further triggers the studies of complicated video understanding tasks, such as video captioning (i.e., generating textual descriptions of videos) [13, 47], video question answering (VQA, i.e., answering questions based on given videos) [23, 61], and video summarization and storytelling [32, 33, 51]. Recently, many video generation methods have been proposed and achieved encouraging performance, e.g., SORA and other related work [18, 40], indicating that the GPT-driven generative models may own strong capability of video semantic understanding in the human level.

As one of the most challenging video understanding tasks, trailer generation selects impressive shots based on understanding the video content and reorganizes the selected shots in an attractive

way. The early methods mainly select and sort video shots based on the utilization of side information. To our knowledge, the work in [29] proposes the first user attention model for trailer generation. Early work in [20, 41] intends to identify impressive audio-visual components by affective content analysis to aid in trailer generation. The work in [1] selects trailer moments in soap operas by combining visual and dialogue information. With the development of machine learning techniques, some learning-based methods are proposed to generate trailers under the guidance of music and texts. The work in [57] considers background music in trailer generation and presents a visual attractiveness model based on point process theory. The work in [26] depends on emotion categories to align images, text, and audio in latent space, selecting and reorganizing video shots by maximizing emotion score.

However, the above methods do not consider the semantic consistency between visual and acoustic embeddings, and they heavily rely on videos with detailed annotations or side information, such as frame-level fixation scores, manually defined emotion labels, and so on. What's more, the current movie-trailer datasets [11, 19, 49] are far from satisfactory. They neither contain multiple trailers corresponding to one movie to achieve conditional learning nor have fine-grained annotations for useful supervision. As a result, the current learning-based trailer generators often suffer from severe overfitting issues. Motivated by the scarcity of high-quality data, we built a movie-trailer dataset with detailed annotations and abundant metadata in this study.

2.2 Optimal transport for matching

As a valid metric of probability measures, optimal transport (OT) [46] has been widely used for distribution comparison and matching. In particular, given two distributions defined in a sample space, we can measure their discrepancy and infer the correspondence between their samples by solving an optimal transport problem and deriving an optimal transport plan (or called coupling [46]) between them accordingly. Therefore, many machine learning tasks that involve matching problems can be modeled as optimal transport problems, e.g., domain adaptation [2, 58], graph matching [4, 37, 56], point cloud registration [38, 39], cross-modal alignment [3, 15, 24], and so on, which all achieve promising results. Typically, the OT problem aims to derive an optimal transport plan to minimize the transport costs between two distributions, which corresponds to a linear programming problem. To solve the problem efficiently (and approximately), Sinkhorn-scaling algorithm [6], proximal point method [53], and Bregman alternating direction method of multipliers (BADMM) [27, 48, 54] are proposed and greatly alleviate the computational complexity of the problem.

Recently, inverse optimal transport (IOT) has been proposed, which aims to optimize the grounding distances associated with samples or their latent representations given observed optimal transport plans [5, 25, 42]. The IOT problem leads to a new learning paradigm to solve a set of latent representation and matching problems, which has been used in many applications. The work in [59] proposes an IOT-based model called IOT-Match for legal case matching, which can generate natural language explanations for matched legal cases and is robust to label insufficiency. The

work in [51] proposes to learn a projection layer to achieve semantic alignment between visual and textual representations via IOT techniques in a self-supervised setting. Typically, the IOT problem corresponds to a bi-level optimization problem, which can be solved effectively by the hypergradient method in [28, 52]. In this study, the correspondence between movies and trailers in our dataset can naturally be seen as observed OT plans, which motivates us to propose the inverse partial optimal transport framework for music-guided trailer generation.

3 Proposed Method

3.1 Problem statement and modeling principle

Suppose that we have a set of movies and their corresponding trailers, denoted as $\mathcal{D} = \{(\mathcal{M}_n, \mathcal{V}_n, \mathcal{A}_n, T_n)\}_{n=1}^N$. Here, $\mathcal{M}_n = \{m_{i,n}\}_{i=1}^{I_n}$ represents the I_n shots of the n -th movie, which corresponds to different scenes happening in the movie. $\mathcal{V}_n = \{v_{j,n}\}_{j=1}^{J_n}$ and $\mathcal{A}_n = \{a_{j,n}\}_{j=1}^{J_n}$ represents the n -th trailer, which contains J_n video and audio shots segmented according to the timestamps of different scenes. In general, the trailer shots are selected from the movie, so we can construct an alignment matrix $\mathcal{T}_n = [t_{ij,n}] \in \{0, 1\}^{I_n \times J_n}$, where $t_{ij,n} = 1$ indicates that the j -th trailer shot corresponds to the i -th movie shot. Obviously, this alignment matrix T_n also works as a movie-music shot map, providing the correspondence across the visual and acoustic modalities, and accordingly, $\mu_n = T_n \mathbf{1}_{J_n} \in \{0, 1\}^{I_n}$ indicates which movie shots are selected to generate the trailer.

In this study, given a movie \mathcal{M} and a piece of music \mathcal{A} , we aim to generate a trailer \mathcal{V} for the movie. We can formulate this music-guided movie trailer generation task as selecting and sorting movie shots conditioned on the music shots, which corresponds to predicting the alignment matrix T between \mathcal{M} and \mathcal{A} (and the associated movie shot indicator μ). In the following content, we will show that when the above dataset is available, we can learn a multi-modal representation and matching model (as illustrated in Figure 1) in a supervised way to achieve this aim, leading to the proposed inverse partial optimal transport framework.

3.2 Model architecture

3.2.1 Multi-modal Self-attentive latent representation. In this study, for an arbitrary movie with I shots and its corresponding trailer with J shots, we first apply the pretrained ImageBind [14] to extract initial embeddings of the movie shots and trailer music shots, respectively, i.e., $\mathbf{M} = f_M(\mathcal{M}) = [\mathbf{m}_i] \in \mathbb{R}^{I \times D}$ and $\mathbf{A} = f_A(\mathcal{A}) = [\mathbf{a}_j] \in \mathbb{R}^{J \times D}$. To make the embedding adaptive to our task and take the temporal correlation of the shots into account, we pass the embeddings of the two modalities through two multi-layer perceptrons (MLPs) and further encode each modalities' embedding by two self-attention (SA) modules, i.e.,

$$\begin{aligned} \mathbf{M}' &= \text{MLP}_M(\mathbf{M}), \quad \mathbf{A}' = \text{MLP}_A(\mathbf{A}), \\ \mathbf{M}^s &= \text{Softmax}\left(\frac{(\mathbf{M}'\mathbf{W}_1^m)(\mathbf{M}'\mathbf{W}_2^m)^\top}{\sqrt{D}}\right)\mathbf{M}'\mathbf{W}_3^m, \\ \mathbf{A}^s &= \text{Softmax}\left(\frac{(\mathbf{A}'\mathbf{W}_1^a)(\mathbf{A}'\mathbf{W}_2^a)^\top}{\sqrt{D}}\right)\mathbf{A}'\mathbf{W}_3^a, \end{aligned} \quad (1)$$

where $\text{MLP}_M, \text{MLP}_A : \mathbb{R}^D \mapsto \mathbb{R}^D$, $\{\mathbf{W}_i^m, \mathbf{W}_i^a \in \mathbb{R}^{D \times D}\}_{i=1}^3$, and $\mathbf{M}^s = [\mathbf{m}_i^s] \in \mathbb{R}^{I \times D}$ and $\mathbf{A}^s = [\mathbf{a}_j^s] \in \mathbb{R}^{J \times D}$ are the proposed latent representations of movie shots and trailer music shots, respectively. The MLPs together with the self-attention modules lead to a multi-modal encoder with a two-tower architecture. As aforementioned, based on the latent representations, we would like to select key movie shots and align them with the trailer music shots, which is achieved by the following two modules.

3.2.2 Cross-attention movie shot selector. Our trailer generator needs to select key movie shots conditioned on given music. Therefore, we propose a cross-attention movie shot selector to predict which movie shots should be selected. In particular, we utilize a cross-attention (CA) module to capture the interactions between the visual and acoustic latent representations, i.e.,

$$\begin{aligned}\bar{\mathbf{M}} &= \mathbf{M}^s + \text{Softmax}\left(\frac{(\mathbf{M}^s \mathbf{W}_4^m)(\mathbf{A}^s \mathbf{W}_4^a)^\top}{\sqrt{D}}\right) \mathbf{A}^s \mathbf{W}_5^a, \\ \bar{\mathbf{A}} &= \mathbf{A}^s + \text{Softmax}\left(\frac{(\mathbf{A}^s \mathbf{W}_5^m)(\mathbf{M}^s \mathbf{W}_6^a)^\top}{\sqrt{D}}\right) \mathbf{M}^s \mathbf{W}_6^m,\end{aligned}\quad (2)$$

where $\{\mathbf{W}_i^m, \mathbf{W}_i^a \in \mathbb{R}^{D \times D}\}_{i=4}^6$, $\bar{\mathbf{M}} = [\bar{\mathbf{m}}_i] \in \mathbb{R}^{I \times D}$ and $\bar{\mathbf{A}} = [\bar{\mathbf{a}}_j] \in \mathbb{R}^{J \times D}$ are the final latent representations of the two modalities. Passing $\bar{\mathbf{M}}$ through the following MLP results in a vector indicating the probabilities of selecting different movie shots, i.e.,

$$\hat{\boldsymbol{\mu}} = [\hat{\mu}_i] = \text{Sigmoid}(\text{MLP}(\bar{\mathbf{M}})) \in [0, 1]^I, \quad (3)$$

where each $\hat{\mu}_i$ indicates the probability that the i -th movie shot is selected to generate a trailer.

3.2.3 Sinkhorn-based movie-music aligner. Besides selecting key movie shots, we need to determine the order of the selected movie shots and make them aligned to the music shots. In this study, we propose a Sinkhorn matching network as the movie-music aligner. In particular, given the visual and acoustic latent representations $\bar{\mathbf{M}}$ and $\bar{\mathbf{A}}$, we can construct a distance matrix $\mathbf{D} = [d(\bar{\mathbf{m}}_i, \bar{\mathbf{a}}_j)] \in \mathbb{R}^{I \times J}$, whose element $d(\bar{\mathbf{m}}_i, \bar{\mathbf{a}}_j)$ represents the Euclidean distance between the latent representation of the i -th movie shot and that of the j -th music shot. The Sinkhorn matching network achieves the cross-modal alignment of the latent representations by solving the following entropic optimal transport (EOT) problem:

$$\hat{\mathbf{T}} = \arg \min_{\mathbf{T} \in \Pi(\frac{1}{\|\hat{\boldsymbol{\mu}}\|_1} \hat{\boldsymbol{\mu}}, \boldsymbol{\gamma})} \underbrace{\langle \mathbf{D}, \mathbf{T} \rangle}_{\mathbb{E}_{\mathbf{T}}[d(\bar{\mathbf{m}}, \bar{\mathbf{a}})]} + \lambda \underbrace{\langle \mathbf{T}, \log \mathbf{T} \rangle}_{\text{Entropy reg.}}, \quad (4)$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product of matrix, $\Pi(\frac{1}{\|\hat{\boldsymbol{\mu}}\|_1} \hat{\boldsymbol{\mu}}, \boldsymbol{\gamma}) = \{\mathbf{T} \geq 0 | \mathbf{T} \mathbf{1}_J = \frac{1}{\|\hat{\boldsymbol{\mu}}\|_1} \hat{\boldsymbol{\mu}}, \mathbf{T}^\top \mathbf{1}_I = \boldsymbol{\gamma}\}$ is the set of the doubly-stochastic matrix, whose marginals must be on the Simplex, i.e., $\frac{1}{\|\hat{\boldsymbol{\mu}}\|_1} \hat{\boldsymbol{\mu}} \in \Delta^{I-1}$ and $\boldsymbol{\gamma} \in \Delta^{J-1}$. As shown in (4), the optimal solution $\hat{\mathbf{T}}$ is called optimal transport plan, which actually is the optimal distribution of latent representation pairs that minimizes the expectation of the distance $d(\bar{\mathbf{m}}, \bar{\mathbf{a}})$, whose marginal distributions are $\frac{1}{\|\hat{\boldsymbol{\mu}}\|_1} \hat{\boldsymbol{\mu}}$ and $\boldsymbol{\gamma}$, respectively. Here, $\frac{1}{\|\hat{\boldsymbol{\mu}}\|_1} \hat{\boldsymbol{\mu}}$ determines the distribution of movie shots, and we set it as the normalized probabilities predicted by the movie shot selector. This setting ensures the predicted alignment result is consistent with the selection of movie shots. On the other hand, because each music shot is applied, we can simply set $\boldsymbol{\gamma}$ to be

Algorithm 1 SinkhornNet($\mathbf{D}, \frac{1}{\|\hat{\boldsymbol{\mu}}\|_1} \hat{\boldsymbol{\mu}}, \boldsymbol{\gamma}; \lambda$)

- 1: Initialize $\mathbf{K} = \exp(-\mathbf{D}/\lambda)$ and $\mathbf{a} = \mathbf{1}$.
 - 2: **While** not converge **do**
 - 3: $\mathbf{b} \leftarrow \frac{\boldsymbol{\gamma}}{\mathbf{K}^\top \mathbf{a}}$ and then $\mathbf{a} \leftarrow \frac{\hat{\boldsymbol{\mu}}}{\mathbf{I} \mathbf{K} \mathbf{b}}$.
 - 4: **return** $\hat{\mathbf{T}} = \text{diag}(\mathbf{a}) \mathbf{K} \text{diag}(\mathbf{b})$
-

uniform, i.e., $\boldsymbol{\gamma} = \frac{1}{J} \mathbf{1}_J$. Finally, the entropic regularizer of the OT plan improves the smoothness of the problem, whose significance is controlled by the hyperparameter $\lambda > 0$.

The EOT problem can be solved efficiently by the Sinkhorn-scaling algorithm shown in Algorithm 1, leading to the implementation of the Sinkhorn matching network with computational complexity $O(IJ)$. Note that the whole algorithmic process is differentiable for both $\hat{\mathbf{T}}$ and the distance matrix \mathbf{D} [28, 52], making the backpropagation applicable in the training phase. As a result, given $\hat{\mathbf{T}} = [\hat{t}_{ij}]$, we can select and align movie shots according to the music shots, i.e., $\hat{i} = \arg \max_{i \in \{1, \dots, I\}} \hat{t}_{ij}$ for $j = 1, \dots, J$.

- **Remark.** It should be noted that compared with selecting and aligning movie shots based on the distance matrix \mathbf{D} (i.e., $\hat{i} = \arg \min_{i \in \{1, \dots, I\}} d(\bar{\mathbf{m}}_i, \bar{\mathbf{a}}_j)$ for $j = 1, \dots, J$), the Sinkhorn matching network often provides better alignment results. In particular, without any constraint, the distance-based alignment may select the same movie shot to match with multiple music shots, which does harm to the diversity of the generated trailer and thus is undesired in practice. On the contrary, the doubly stochastic constraint encourages the optimal transport plan $\hat{\mathbf{T}}$ to achieve the one-one correspondence between the movie and music shots.

3.3 Inverse partial optimal transport framework

3.3.1 The IPOT-based supervised learning paradigm. Denote θ as the model parameters in the MLPs and the self- and cross-attention modules. When the dataset $\mathcal{D} = \{\mathcal{M}_n, \mathcal{V}_n, \mathcal{A}_n, \mathcal{T}_n\}_{n=1}^N$ is available, we learn our model in a supervised way by solving the following inverse partial optimal transport (IPOT) problem:

$$\begin{aligned}\min_{\theta} \quad & \sum_{n=1}^N \underbrace{\text{KL}(\hat{\mathbf{T}}_n(\theta) \parallel \frac{1}{J_n} \mathbf{T}_n)}_{\text{Supervision of Aligner}} + \delta \underbrace{\text{BCE}(\hat{\boldsymbol{\mu}}_n(\theta), \boldsymbol{\mu}_n)}_{\text{Supervision of Selector}} \\ \text{s.t.} \quad & \underbrace{\hat{\mathbf{T}}_n(\theta) = \arg \min_{\mathbf{T} \in \Pi(\boldsymbol{\mu}_n, \boldsymbol{\gamma}_n)} \langle \mathbf{D}_n(\theta), \mathbf{T} \rangle + \lambda \langle \mathbf{T}, \log \mathbf{T} \rangle}_{\text{Entropic Partial Optimal Transport}} \\ & \forall n = 1, \dots, N.\end{aligned}\quad (5)$$

As shown in (5), the IPOT problem is a bi-level optimization problem. In the upper-level problem, given each observed alignment matrix \mathbf{T}_n , we take its normalized version $\frac{1}{J_n} \mathbf{T}_n$ as the ground truth optimal transport plan between \mathcal{M}_n and \mathcal{A}_n and supervise the learning of our movie-music aligner. $\boldsymbol{\mu}_n = \mathbf{T}_n \mathbf{1}_{J_n}$ denotes the ground truth of movie shot selection, which supervises the learning of our movie shot selector. For each movie-music pair, the first term in the upper-level problem penalizes the KL-divergence between the predicted alignment result $\hat{\mathbf{T}}_n(\theta)$ and the ground truth $\frac{1}{J_n} \mathbf{T}_n$. The second term penalizes the binary cross-entropy (BCE) loss between the

predicted selection probabilities $\hat{\mu}_n(\theta)$ and the ground truth μ_n . $\delta > 0$ controls the trade-off between the two terms.

The constraint of the upper-level problem corresponds to the lower-level optimization problem deriving the optimal transport plan. Compared with the EOT problem in (4), the lower-level problem in (5) takes the ground truth μ_n as the marginal distribution directly. Because of the sparsity of μ_n , this problem is formulated as an entropic partial optimal transport problem — the rows of $\hat{T}_n(\theta)$ corresponding to those unselected movie shots are set to be all-zeros so that we do not need to consider them during training. Such a strategy helps to decouple the learning objectives in the upper-level problem. In particular, while the BCE term focuses on learning the movie shot selector, by filtering out unselected movie shots, the KL-divergence term in the upper-level problem supervises the learning of the movie-music aligner for the selected movie shots and the music shots, which avoids the unnecessary mismatching with unselected movie shots.

3.3.2 Learning algorithm. This IPOT problem can be solved efficiently by a stochastic gradient descent (SGD) algorithm. Given a batch of movie-music pairs, i.e., $\mathcal{B} \subset \mathcal{D}$, we first solve a set of entropic partial optimal transport problems and obtain the optimal transport plans for each movie-music pair, i.e., for $n \in \mathcal{B}$, we obtain $\hat{T}_n(\theta) = \text{SinkhornNet}(D_n(\theta), \mu_n, \frac{1}{J} \mathbf{1}_J; \lambda)$. Then, we update the model parameters using SGD. Denote the objective function of the upper-level problem corresponding to the batch as $L(\theta)$. When computing the gradient of $L(\theta)$, we leverage the hypergradient method in [28, 53], i.e.,

$$\nabla_{\theta} L(\theta) = \sum_{n \in \mathcal{B}} \frac{\partial L(\theta)}{\partial D_n(\theta)} \frac{\partial D_n(\theta)}{\partial \theta} + \frac{\partial L(\theta)}{\partial \hat{T}_n(\theta)} \frac{\partial \hat{T}_n(\theta)}{\partial \theta}, \quad (6)$$

in which the second term involves the hypergradient term $\frac{\partial \hat{T}_n(\theta)}{\partial \theta}$, which requires us to unroll the Sinkhorn-scaling iterations in Algorithm 1. In general, this hypergradient term can be derived either by auto-differentiation [16, 55]. In our case, because $\hat{T}_n(\theta)$ is the solution of an entropic optimal transport problem, this term can also be derived in a closed form. Please refer to Theorem 2 in [52] for more details. The remaining terms in (6) are derived by backpropagation.

3.4 Trailer generation pipeline

Given a well-trained model θ^* , we can achieve music-guided movie trailer generation by an efficient pipeline. In particular, given a movie, we first resize it to 320p and then apply the video segmentation tool BaSSL [31] to obtain movie shots, i.e., $\mathcal{M} = \{m_i\}_{i=1}^I$. When a piece of music is provided, we first use the Ultimate Vocal Remover (UVR) tool to eliminate the vocal part, leaving only the background track, and then obtain music shots by the music segmentation tool Ruptures [45] method, i.e., $\mathcal{A} = \{a_j\}_{j=1}^J$. As aforementioned, both the movie and music shots are initially embedded by a pre-trained ImageBind [14].

By utilizing the well-trained movie shot selector and the latent representation model, we calculate the probability vector $\hat{\mu}(\theta^*)$ for movie shots and select the shots with J highest probabilities to construct the trailer, i.e., $\mathcal{V} = \arg \text{sort-} J_{i \in \{1, \dots, I'\}} \hat{\mu}_i$. Here, $I' = 0.9I$, which means that we only consider the first 90% of movie shots instead of all shots for spoiler prevention. After deriving the final

| Dataset | LSMTD [19] | TMDD [49] | MovieLights [11] | CMTD |
|---------------|------------|-----------|------------------|------|
| #Movie | 508 | 150 | 174 | 208 |
| #Trailer | 34,219 | 150 | 174 | 406 |
| Multi-trailer | ✗ | ✗ | ✗ | ✓ |
| Segments | ✗ | ✗ | ✓ | ✓ |
| Alignment | ✗ | ✗ | ✓ | ✓ |
| Metadata | ✗ | ✗ | ✗ | ✓ |

Table 1: A comparison for various datasets. We use green ticks to mark the information used for trailer generation.

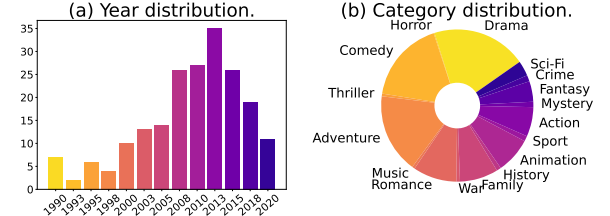


Figure 2: Visualization of the publication year distribution and the category proportions of movies in CMTD.

latent representations of selected movie shots and music shots, i.e., $\bar{V} = [\bar{v}_j] \in \mathbb{R}^{J \times D}$ and $\bar{A} = [\bar{a}_j] \in \mathbb{R}^{J \times D}$, we infer the one-one correspondence between them by solving an EOT problem:

$$\hat{T} = \arg \min_{T \in \Pi(\frac{1}{J} \mathbf{1}_J, \frac{1}{J} \mathbf{1}_J)} \langle D, T \rangle + \lambda \langle T, \log T \rangle, \quad (7)$$

where the distance matrix $D = [d_{ij}] \in \mathbb{R}^{J \times J}$ contains the discrepancy between each selected movie shot and each music shot. In this study, we define its elements as

$$d_{ij} = \underbrace{\|\bar{v}_i - \bar{a}_j\|_2^2}_{\text{Semantic dis.}} + \eta \underbrace{|\tau_i^m - \tau_j^a|}_{\text{Temporal dis.}}, \quad \forall i, j = 1, \dots, J. \quad (8)$$

Here, for the i -th selected movie shot and the j -th music shot, the first term in (8) indicates the semantic discrepancy between their latent representations, while the second term in (8) indicates their temporal discrepancy, where τ_i^m and τ_j^a denote the lengths of the two shots. Typically, it is easy to synchronize the two shots when $|\tau_i^m - \tau_j^a|$ is small. The hyperparameter $\eta > 0$ achieves the trade-off between the two terms. Note that, in the training phase, we do not consider the temporal discrepancy when constructing the $D_n(\theta)$'s in (5) because the aligned shots in our training data have the same duration while those unaligned shots often have different lengths. Introducing the temporal discrepancy would oversimplify the learning task and weaken the supervision on our model.

After inferring the aligned shot pairs based on \hat{T} , we engage in post-processing the aligned movie shots to adapt the duration of the music shots. For each music shot, when the corresponding movie shot exceeds its duration, we cut the movie shot to match its duration. When the corresponding movie shot falls short on length, we extend the movie shot by incorporating one or more adjacent movie shots based on their probabilities (i.e., $\hat{\mu}(\theta^*)$). Finally, we concatenate the post-processed movie shots and take the music as the soundtrack, composing the ultimate movie trailer.

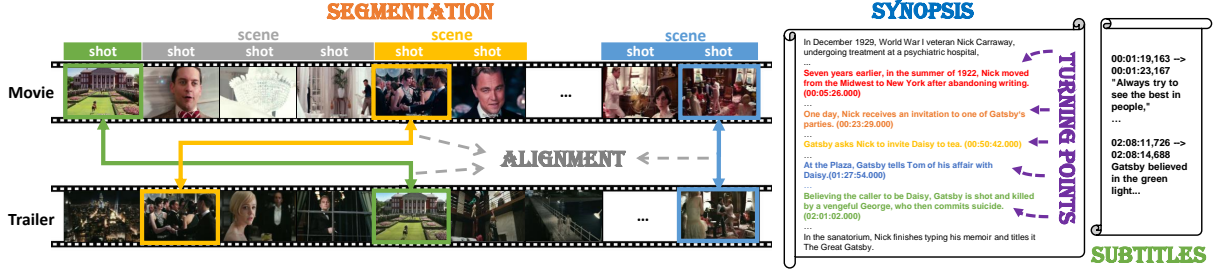


Figure 3: An illustration of the annotations associated with the movie "The Great Gatsby" in CMTD.

4 The CMTD Dataset for Training

Implementing our IPOT learning framework needs a movie-trailer dataset with detailed annotations (e.g., the segmentation and alignment information). Unfortunately, existing datasets, e.g., Large-Scale Movie and Trailer Dataset (LSMTD) [19], Trailer Momont Detection Dataset (TMDD) [49] and Movie Highlight Detection Dataset (MovieLights) [11], are non-public and fail to meet our requirement. The movies and trailers in LSMTD are not paired. Although the movies and trailers in TMDD and MovieLights are paired, each movie is only associated with a single trailer (and its music). In the music-guided trailer generation task, we expect that each movie corresponds to multiple trailers, which helps suppress the risk of over-fitting. The above problems motivate us to build our Comprehensive Movie-Trailer Dataset, called CMTD for short.

4.1 Data collection and annotation

As shown in Figure 2, CMTD contains 208 movies and 406 trailers. These movies and trailers have sufficient richness and diversity in content and year, which are categorized into 18 classes based on their tags at IMDB. Each movie corresponds to one to six trailers, roughly two trailers per movie on average. The average duration per movie and trailer is 1.91 hours and 2.17 minutes, respectively. We apply BaSSL [31] to segment each movie/trailer into shots and aggregate adjacency shots in the scene level. The average shot number and scene number per movie are 1909 and 57, respectively.

To obtain the alignment matrix automatically with high accuracy, for each movie-trailer pair, we first obtain frame-level visual embeddings for the movie and the trailer by ImageBind [14]. Based on the embeddings, we then apply Faiss [8, 21] to efficiently compute the visual similarity between the trailer's frames and the movie's. For each trailer frame, we can get the top-4 movie frames that most closely resemble it. For a trailer shot having K frames, we can select $4K$ most similar movie shots — each trailer frame is matched with four movie shots where the corresponding top-4 movie frames belong to. Among the $4K$ movie shots, the shot with the highest number of occurrences is annotated as the correspondence of the trailer shot. Applying this annotation method to all trailer shots, we can calculate the alignment matrix T between the movie and trailer shots. By random sampling and manual verification, we confirm the reliability of this annotation method.

Besides the shot-level alignment information, CMTD also provides abundant auxiliary information as metadata, including subtitles, synopsis, turning points annotations, and so on. In particular,

for each movie, we collect its subtitle from Subscene and collect its synopsis from the movie's Wikipedia page. Based on the synopsis, we further annotate five turning points (i.e., the key moments in the storyline) that define the narrative structure of the movie [35, 36]. Figure 3 illustrates the annotations associated with a movie, and Table 1 shows the comparison among different datasets. Note that, although we just apply the segment and alignment information in the trailer generation task (for a fair comparison with baseline methods), the metadata in CMTD can support more applications and thus contribute to promoting more studies of video understanding.

4.2 Data release plan and its social impacts

We plan to make our CMTD dataset public. To achieve a trade-off between the acceleration of research and the protection of intellectual property, we are considering the following strategies.

- We plan to develop a license agreement that further stipulates the use scope and limitations, including prohibiting redistribution, commercial use, modification, etc., to ensure the dataset is used only for non-commercial research and academic purposes. Before accessing our data, each user is required to submit a signed application form and provide his/her education email, promising to obey the agreement.
- For all movies, trailers, and music, we plan to release their embeddings and annotations, including those metadata, such that if the users can access the raw videos, they can easily segment and align the videos based on the annotations. Additionally, for the convenience of research, we also consider releasing movies and trailers with extremely low resolutions and/or watermarks, preventing them from being used for other purposes except research.

We expect CMTD to be the first publicly available movie-trailer dataset, advancing the academic field of video understanding and triggering more interesting and significant research work.

5 Experiments

To demonstrate the effectiveness of our IPOT-based trailer generator, we compare it with state-of-the-art methods through both objective and subjective evaluations. **More details and results can be found in the arXiv version.**

5.1 Implementation Details

5.1.1 Baselines. We take state-of-the-art trailer generation methods as baselines, including V2T [20], M2T [43], and PPBVAM [57].

| Category | Method | Movie shot selection | | | | | | | | | Movie-music alignment | | | |
|--------------------|--------------|----------------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|-----------------------|---------------|---------------|----------------|
| | | P@1↑ | P@3↑ | P@5↑ | R@1↑ | R@3↑ | R@5↑ | F1@1↑ | F1@3↑ | F1@5↑ | P@1↑ | R@1↑ | F1@1↑ | KL↓ |
| Video Summary | VASNet [9] | 0.0237 | 0.0725 | 0.1102 | 0.0343 | 0.1096 | 0.1698 | 0.0277 | 0.0861 | 0.1317 | — | — | — | — |
| | Muvee [12] | 0.2130 | 0.3245 | 0.3452 | 0.0414 | 0.0612 | 0.0690 | 0.0640 | 0.0949 | 0.1059 | — | — | — | — |
| | CLIP-It [33] | 0.0302 | 0.0863 | 0.1468 | 0.0527 | 0.1409 | 0.2429 | 0.0378 | 0.1054 | 0.1801 | — | — | — | — |
| | OTVS [51] | 0.0637 | 0.1398 | 0.1821 | 0.0941 | <u>0.2157</u> | <u>0.2864</u> | 0.0746 | <u>0.1669</u> | <u>0.2193</u> | — | — | — | — |
| Trailer Generation | M2T [43] | 0.0229 | 0.0347 | 0.0444 | 0.0188 | 0.0273 | 0.0362 | 0.0193 | 0.0285 | 0.0371 | 0.0028 | 0.0031 | 0.0029 | 2417.71 |
| | V2T [20] | 0.0787 | 0.1397 | 0.2031 | 0.0396 | 0.0693 | 0.1035 | 0.0508 | 0.0891 | 0.1322 | <u>0.0028</u> | <u>0.0031</u> | <u>0.0029</u> | <u>1855.59</u> |
| | PPBVAM [57] | 0.0687 | 0.1339 | 0.1862 | <u>0.1003</u> | 0.2000 | 0.2729 | <u>0.0781</u> | 0.1537 | 0.2117 | 0.0019 | 0.0022 | 0.0020 | 2871.03 |
| | IPOT (Ours) | <u>0.1098</u> | <u>0.2248</u> | <u>0.3064</u> | 0.1234 | 0.2536 | 0.3446 | 0.1161 | 0.2381 | 0.3240 | 0.0075 | 0.0081 | 0.0078 | 1696.42 |

Table 2: Comparisons on movie shot selection. We bold the best results and underline the second-best results.

When evaluating the movie shot selector learned by our method, we also compare it with three state-of-the-art video summarization methods (i.e., VASNet [9], CLIP-It [33], and OTVS [51]) and one commercial video summarization software Muvee [12]. For learning-based methods, including ours, we select 200 movies from CMTD for training and apply the remaining eight movies for evaluation. Note that, because most of the baselines only release trailers generated from the eight movies rather than their code, we select the eight movies for a fair comparison.

5.1.2 Evaluation Metrics. For the ground truth trailer and the generated one, we can find the indices of their shots in the corresponding movie and construct two index sequences, denoted as $A = \{\alpha_1, \dots, \alpha_I\}$ and $B = \{\beta_1, \dots, \beta_J\}$. Therefore, when evaluating our movie shot selector, we take three commonly used metrics: Top- K Precision, Recall, and F1-Score, i.e., $P@K = |A \cap_K B| / |A|$, $R@K = |A \cap_K B| / |B|$, and $F1@K = \frac{2P@K \cdot R@K}{P@K + R@K}$, where $|\cdot|$ is the cardinality of set, $K = 1, 3, 5$, and $A \cap_K B = \{i | |\alpha_i - \beta_i| \leq K - 1\}$ counts the number of shot pairs with close enough indices.

For each trailer generator, we use P@1, R@1, F1@1, and KL-divergence between the estimated and observed alignment matrices to quantitatively evaluate the alignment between movie shots and audio shots. Additionally, the statistics of trailer shots is applied to evaluate the quality of generated trailers as well – we record the number and average duration of trailer shots generated by each method and compare these values with those of official trailer shots. Besides objective measurements, we also evaluate different methods through subjective user study.

5.1.3 Model and hyperparameter settings. The self-attention and cross-attention modules are implemented as Transformer encoders, each of which has one layer and two attention heads. There are linear layers both before and after the two types of attention modules. When training our model, we apply Adam [22] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate is $1e-5$ and the training epoch is 500.

5.2 Quantitative and Qualitative Comparisons

Table 2 shows the performance of various methods on movie shot selection. We can see that our IPOT-based method achieves the best performance on most measurements. Especially in terms of F1-Scores, our method works best in all three settings. These results demonstrate the superiority of our method on movie shot selection – it is more likely to select the movie shots that are used in official trailers. In the aspect of movie-music shot alignment, we mainly

| Methods | Test movie-1 | | Test movie-6 | |
|------------------|--------------|-------|--------------|-------|
| | Duration (s) | #Shot | Duration (s) | #Shot |
| Official Trailer | 1.95±1.82 | 77 | 2.35±2.91 | 63 |
| PPBVAM [57] | 1.12±0.39 | 163 | 1.26±0.46 | 131 |
| Muvee [12] | 7.72±8.73 | 24 | 42.69±33.83 | 4 |
| V2T [20] | 4.06±5.72 | 44 | 2.83±2.53 | 58 |
| M2T [43] | 1.71±0.83 | 89 | 1.72±0.75 | 89 |
| IPOT (Ours) | 2.03±2.00 | 74 | 2.01±1.73 | 58 |

Table 3: Comparisons on trailer shot number and duration.

compare our method with other trailer generators. The results in Table 2 show that our method can achieve the highest precision, recall, and F1-score and the lowest KL-divergence, which means that the alignment achieved by the OT plan matches better with the ground truth than other methods. Figure 4 provides an example comparing the generated trailers of different methods with the official one, which visualizes the advantage of our method.

Table 3 shows the comparison on the number and average duration of trailer shots generated by various methods. The number of shots in different trailers is distinct. Some methods choose a very small number of shots in a trailer, such as Muvee, making these methods achieve high precision but low recall. On the contrary, some methods choose a massive number of trailer shots, such as PPBVAM [57], making them dominant in recall. Our method is more balanced, whose number and duration of trailer shots are close to those of official trailer shots, so that it achieves the best performance on F1-scores.

5.3 Subjective User Study

Besides objective evaluation, we evaluate our method as well as the baselines (i.e., V2T, M2T, PPBVAM) through subjective user studies, comparing their user scores with those of official trailers (RT). Following the work in [20, 57], we propose to compare different trailers in the following five aspects:

- **Character:** How does the trailer include close-up shots of the main characters in the movie?
- **Rhythm:** How well does the montage match the rhythm of the background music?
- **Attractiveness:** How attractive is the trailer? How much are you impressed by this trailer?
- **Appropriateness:** How close is the trailer to a real trailer?
- **Interest:** How interested do you become in watching this movie after watching the trailer?



Figure 4: Comparison between some generated trailer shots and the official trailer shots of the movie "Elysium" based on their appearance order. For each generated trailer, their correctly selected shots are marked with green boxes. The selected shot of our trailer connected to the shot in the official trailer by a yellow arrow means that they belong to the same scene.

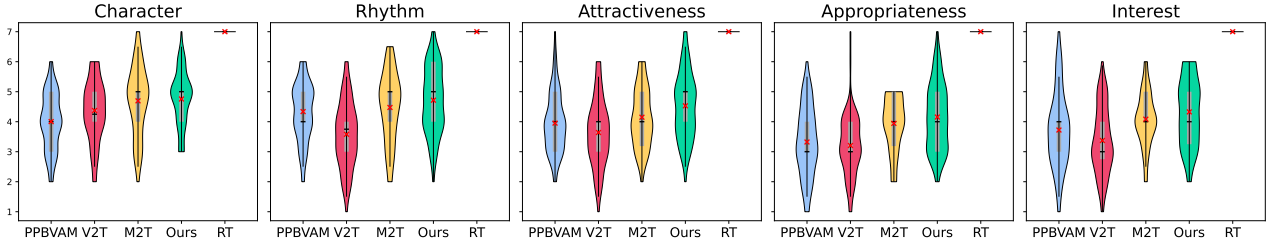


Figure 5: The violin plot of scores for various methods in user studies. The red crosses are means and the black bars are medians.

| Setting | Selection | | | Alignment | |
|-----------------|---------------|---------------|---------------|---------------|----------------|
| | F1@1↑ | F1@3↑ | F1@5↑ | F1@1↑ | KL↓ |
| -w/o SA | 0.0280 | 0.0815 | 0.1015 | 0.0011 | 1706.50 |
| -w/o CA | 0.0317 | 0.0907 | 0.1454 | 0.0020 | 1704.48 |
| -w/o PartialOT | 0.1077 | 0.1536 | 0.1976 | 0.0036 | 1702.47 |
| Proposed | 0.1129 | 0.2178 | 0.3240 | 0.0078 | 1696.42 |

Table 4: Ablation study on the model components.

All trailers are processed to the same resolution (320×240). Given the trailers generated by different methods, we establish a website and invite 25 volunteers (7 females and 18 males) to watch them, in which the names of the methods are anonymous and the order of the trailers on the website is random. For each movie, a volunteer scores the corresponding generated trailers from one (the lowest) to seven (the highest) in each of the above five aspects, where the score of the official trailer is set to be seven by default. Figure 5 shows the results of various methods. On average, our method consistently outperforms the three baselines in the five aspects.

Note that, because the quality of the movie trailer is finally evaluated by the audience in practice, which is highly subjective, the above user study is necessary and can provide complementary information compared to the objective measurements. For example, the F1-score of M2T in movie shot selection is very low, but its scores in the user study are better than V2T and PPBVAM. This phenomenon implies that it may select relatively reasonable shots that are not used in official trailers.

5.4 Ablation Study

Table 4 displays the results of some ablation experiments, demonstrating the significance of different model components. Removal of the self-attention (SA) mechanism from the framework results in a significant degradation in both selection and alignment performance. This may be attributed to SA's role in establishing temporal relationships among shots and enabling the integration of contextual information. Eliminating the cross-attention (CA) mechanism disrupts the semantic interaction between the two modalities, making their alignment more challenging and leading to a notable decrease in performance. In addition, when training the model without the partial OT mechanism, i.e., replacing the μ_n with $\hat{\mu}_n$ in the lower-level problem of (5), the model performance also degrades because of introducing unnecessary uncertainty.

6 Conclusion

In this work, we propose an inverse partial optimal transport (IPOT) framework for music-guided trailer generation and build a comprehensive movie-trailer dataset to support the learning of the trailer generator. The proposed trailer generator consists of a music-guided movie shot selector and a movie-music shot aligner, which can be learned effectively by a bi-level optimization strategy. Experiments demonstrate that our IPOT-based method outperforms state-of-the-art trailer generation and video summarization methods on both objective and subjective evaluation measurements.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (62102031), and the foundation of Key Laboratory of Artificial Intelligence, Ministry of Education, Shanghai, P.R. China (AI202409).

References

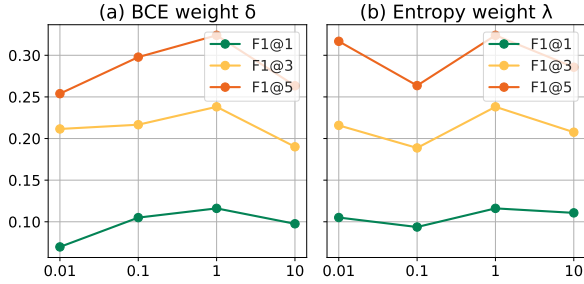
- [1] Carlo Bretti, Pascal Mettes, Hendrik Vincent Koops, Daan Odijk, and Nanne van Noord. 2024. Find the Cliffhanger: Multi-modal Trailerness in Soap Operas. In *International Conference on Multimedia Modeling*. Springer, 199–212.
- [2] Liqun Chen, Zhe Gan, Yu Cheng, Linjie Li, Lawrence Carin, and Jingjing Liu. 2020. Graph optimal transport for cross-domain alignment. In *International Conference on Machine Learning*. PMLR, 1542–1553.
- [3] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning. In *European conference on computer vision*. Springer, 104–120.
- [4] Haoran Cheng, Dixin Luo, and Hongteng Xu. 2023. DHOT-GM: Robust Graph Matching Using A Differentiable Hierarchical Optimal Transport Framework. *arXiv preprint arXiv:2310.12081* (2023).
- [5] Wei-Ting Chiu, Pei Wang, and Patrick Shafto. 2022. Discrete probabilistic inverse optimal transport. In *International Conference on Machine Learning*. PMLR, 3925–3946.
- [6] Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems* 26 (2013).
- [7] Daniel DeMenthon and David Doermann. 2003. Video retrieval using spatio-temporal descriptors. In *Proceedings of the eleventh ACM international conference on Multimedia*. 508–517.
- [8] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The Faiss library. (2024). *arXiv:2401.08281* [cs.LG]
- [9] Jiri Fajtl, Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, and Paolo Remagnino. 2018. Summarizing videos with attention. In *Asian Conference on Computer Vision*. Springer, 39–54.
- [10] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. 2020. Multi-modal transformer for video retrieval. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV* 16. Springer, 214–229.
- [11] Bei Gan, Xiujun Shu, Ruizhi Qiao, Haoqian Wu, Keyu Chen, Hanjun Li, and Bo Ren. 2023. Collaborative Noisy Label Cleaner: Learning Scene-aware Trailers for Multi-modal Highlight Detection in Movies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18898–18907.
- [12] Roman Ganhör. 2014. Muvee: An Alternative Approach to Mobile Video Trimming. In *2014 IEEE International Symposium on Multimedia*. IEEE, 229–236.
- [13] Lianli Gao, Zhao Guo, Hanwang Zhang, Xing Xu, and Heng Tao Shen. 2017. Video captioning with attention-based LSTM and semantic consistency. *IEEE Transactions on Multimedia* 19, 9 (2017), 2045–2055.
- [14] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. 2023. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15180–15190.
- [15] Fengjiao Gong, Yuzhou Nie, and Hongteng Xu. 2022. Gromov-Wasserstein multi-modal alignment and clustering. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 603–613.
- [16] Stephen Gould, Richard Hartley, and Dylan Campbell. 2021. Deep declarative networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 8 (2021), 3988–4004.
- [17] Michael Hauge. 2017. *Storytelling Made Easy: Persuade and Transform Your Audiences, Buyers, And Clients—Simply, Quickly, and Profitably*. BookBaby.
- [18] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. 2022. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303* (2022).
- [19] Qingqiu Huang, Yuanjun Xiong, Yu Xiong, Yuqi Zhang, and Dahua Lin. 2018. From trailers to storylines: An efficient way to learn from movies. *arXiv preprint arXiv:1806.05341* (2018).
- [20] Go Irie, Takashi Satou, Akira Kojima, Toshihiko Yamasaki, and Kiyoharu Aizawa. 2010. Automatic trailer generation. In *Proceedings of the 18th ACM international conference on Multimedia*. 839–842.
- [21] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data* 7, 3 (2019), 535–547.
- [22] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- [23] Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. 2020. Hierarchical conditional relation networks for video question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 9972–9981.
- [24] John Lee, Max Dabagia, Eva Dyer, and Christopher Rozell. 2019. Hierarchical optimal transport for multimodal distribution alignment. *Advances in neural information processing systems* 32 (2019).
- [25] Ruilin Li, Xiaojing Ye, Haomin Zhou, and Hongyuan Zha. 2019. Learning to match via inverse optimal transport. *Journal of machine learning research* 20, 80 (2019), 1–37.
- [26] Wu-Qin Liu, Min-Xuan Lin, Hai-Bin Huang, Chong-Yang Ma, Yu Song, Wei-Ming Dong, and Chang-Sheng Xu. 2023. Emotion-Aware Music Driven Movie Montage. *Journal of Computer Science and Technology* 38, 3 (2023), 540–553.
- [27] Dixin Luo, Yutong Wang, Angxiao Yue, and Hongteng Xu. 2022. Weakly-supervised temporal action alignment driven by unbalanced spectral fused Gromov-Wasserstein distance. In *Proceedings of the 30th ACM International Conference on Multimedia*. 728–739.
- [28] Dixin Luo, Hongteng Xu, and Lawrence Carin. 2023. Differentiable Hierarchical Optimal Transport for Robust Multi-View Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 6 (2023), 7293–7307.
- [29] Yu-Fei Ma, Lie Lu, Hong-Jiang Zhang, and Mingjing Li. 2002. A user attention model for video summarization. In *Proceedings of the tenth ACM international conference on Multimedia*. 533–542.
- [30] Zongyang Ma, Ziqi Zhang, Yuxin Chen, Zhongang Qi, Yingmin Luo, Zekun Li, Chunfeng Yuan, Bing Li, Xiaohu Qie, Ying Shan, et al. 2023. Order-Prompted Tag Sequence Generation for Video Tagging. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 15681–15690.
- [31] Jonghwan Mun, Minchul Shin, Gunsoo Han, Sangho Lee, Seongsu Ha, Joonseok Lee, and Eun-Sol Kim. 2022. Boundary-aware self-supervised learning for video scene segmentation. *arXiv preprint arXiv:2201.05277* (2022).
- [32] Medhini Narasimhan, Arsha Nagrani, Chen Sun, Michael Rubinstein, Trevor Darrell, Anna Rohrbach, and Cordelia Schmid. 2022. TL; DW? Summarizing Instructional Videos with Task Relevance & Cross-Modal Saliency. *arXiv preprint arXiv:2208.06773* (2022).
- [33] Medhini Narasimhan, Anna Rohrbach, and Trevor Darrell. 2021. Clip-it! language-guided video summarization. *Advances in neural information processing systems* 34 (2021), 13988–14000.
- [34] Harrie Oosterhuis, Sujith Ravi, and Michael Bendersky. 2016. Semantic video trailers. *arXiv preprint arXiv:1609.01819* (2016).
- [35] Pinelopi Papalampidi, Frank Keller, and Mirella Lapata. 2019. Movie plot analysis via turning point identification. *arXiv preprint arXiv:1908.10328* (2019).
- [36] Pinelopi Papalampidi, Frank Keller, and Mirella Lapata. 2021. Movie summarization via sparse graph construction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 13631–13639.
- [37] Hermína Petric Maretic, Mireille El Gheche, Giovanni Chierchia, and Pascal Frossard. 2019. GOT: an optimal transport framework for graph comparison. *Advances in Neural Information Processing Systems* 32 (2019).
- [38] Gilles Puy, Alexandre Boulch, and Renaud Marlet. 2020. Flot: Scene flow on point clouds guided by optimal transport. In *European conference on computer vision*. Springer, 527–544.
- [39] Zhengyang Shen, Jean Feydy, Peirong Liu, Ariel H Curiale, Ruben San Jose Estepar, Raul San Jose Estepar, and Marc Niethammer. 2021. Accurate point cloud registration with robust optimal transport. *Advances in Neural Information Processing Systems* 34 (2021), 5373–5389.
- [40] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. 2022. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792* (2022).
- [41] John R Smith, Dhiraj Joshi, Benoit Huet, Winston Hsu, and Jozef Cota. 2017. Harnessing ai for augmenting creativity: Application to movie trailer creation. In *Proceedings of the 25th ACM international conference on Multimedia*. 1799–1808.
- [42] Andrew M Stuart and Marie-Therese Wolfram. 2020. Inverse optimal transport. *SIAM J. Appl. Math.* 80, 1 (2020), 599–619.
- [43] Domen Tabernik, Alan Lukežić, and Klemen Grm. [n. d.]. movie2trailer: Unsupervised trailer generation using Anomaly detection. ([n. d.]).
- [44] Kristin Thompson. 1999. *Storytelling in the new Hollywood: Understanding classical narrative technique*. Harvard University Press.
- [45] Charles Truong, Laurent Oudre, and Nicolas Vayatis. 2020. Selective review of offline change point detection methods. *Signal Processing* 167 (2020), 107299.
- [46] Cédric Villani et al. 2009. *Optimal transport: old and new*. Vol. 338. Springer.
- [47] Bairui Wang, Lin Ma, Wei Zhang, and Wei Liu. 2018. Reconstruction network for video captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7622–7631.
- [48] Huahua Wang and Arindam Banerjee. 2014. Bregman alternating direction method of multipliers. *Advances in Neural Information Processing Systems* 27 (2014).
- [49] Lezi Wang, Dong Liu, Rohit Puri, and Dimitris N Metaxas. 2020. Learning trailer moments in full-length movies with co-contrastive attention. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII* 16. Springer, 300–316.
- [50] Xiao Wang, Tian Gan, Yinwei Wei, Jianlong Wu, Dai Meng, and Liqiang Nie. 2022. Micro-video tagging via jointly modeling social influence and tag relation. In

- Proceedings of the 30th ACM International Conference on Multimedia*. 4478–4486.
- [51] Yutong Wang, Hongteng Xu, and Dixin Luo. 2023. Self-supervised Video Summarization Guided by Semantic Inverse Optimal Transport. In *Proceedings of the 31st ACM International Conference on Multimedia*. 6611–6622.
 - [52] Yujia Xie, Yixiu Mao, Simiao Zuo, Hongteng Xu, Xiaojing Ye, Tuo Zhao, and Hongyuan Zha. 2020. A hypergradient approach to robust regression without correspondence. In *International Conference on Learning Representations*.
 - [53] Yujia Xie, Xiangfeng Wang, Ruijia Wang, and Hongyuan Zha. 2020. A fast proximal point method for computing exact wasserstein distance. In *Uncertainty in artificial intelligence*. PMLR, 433–453.
 - [54] Hongteng Xu. 2020. Gromov-Wasserstein factorization models for graph clustering. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 6478–6485.
 - [55] Hongteng Xu and Minjie Cheng. 2023. Regularized optimal transport layers for generalized global pooling operations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).
 - [56] Hongteng Xu, Dixin Luo, Hongyuan Zha, and Lawrence Carin Duke. 2019. Gromov-wasserstein learning for graph matching and node embedding. In *International conference on machine learning*. PMLR, 6932–6941.
 - [57] Hongteng Xu, Yi Zhen, and Hongyuan Zha. 2015. Trailer generation via a point process-based visual attractiveness model. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.
 - [58] Renjun Xu, Pelen Liu, Yin Zhang, Fang Cai, Jindong Wang, Shuoying Liang, Heting Ying, and Jianwei Yin. 2020. Joint Partial Optimal Transport for Open Set Domain Adaptation.. In *IJCAI*. 2540–2546.
 - [59] Weijie Yu, Zhongxiang Sun, Jun Xu, Zhenhua Dong, Xu Chen, Hongteng Xu, and Ji-Rong Wen. 2022. Explainable legal case matching via inverse optimal transport-based rationale extraction. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*. 657–668.
 - [60] Peisen Zhao, Lingxi Xie, Chen Ju, Ya Zhang, Yanfeng Wang, and Qi Tian. 2020. Bottom-up temporal action localization with mutual regularization. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII* 16. Springer, 539–555.
 - [61] Linchao Zhu, Zhongwen Xu, Yi Yang, and Alexander G Hauptmann. 2017. Uncovering the temporal context for video question answering. *International Journal of Computer Vision* 124 (2017), 409–421.
 - [62] Zixin Zhu, Wei Tang, Le Wang, Nanning Zheng, and Gang Hua. 2021. Enriching local and global contexts for temporal action localization. In *Proceedings of the IEEE/CVF international conference on computer vision*. 13516–13525.

| Turning point | Definition |
|--------------------|-----------------------------------------------------------------------------------------------------------------|
| Opportunity | Introductory event that occurs after the presentation of the setting and the background of the main characters. |
| Change of Plans | Event where the main goal of the story is defined. From this point on, the action begins to increase. |
| Point of No Return | Event that pushes the main character(s) to fully commit to their goal. |
| Major Setback | Event where everything falls apart (temporarily or permanently). |
| Climax | Final event of the main story, moment of resolution and the “biggest spoiler”. |

Table 5: Turning points and their definitions.

| Strategy | F1@1↑ | F1@3↑ | F1@5↑ |
|------------------------------|---------------|---------------|---------------|
| Train on MVs | 0.0857 | 0.2051 | 0.2905 |
| Train on CMTD | 0.0910 | 0.1781 | 0.2427 |
| Pretrain&Finetune | 0.1161 | 0.2381 | 0.3240 |

Table 6: Ablation study to explore the impact of different training strategies on the performance.**Figure 6: The influence of two key hyperparameters.**

A Implementation Details

A.1 Implementation of the baselines

In comparison to the baseline methods for video summarization (e.g., VASNet, CLIP-It, OTVS), which require 512-dimensional CLIP features as input, we re-extract the features of the test movies. Specifically, we treat each shot of the movie as a basic unit. Subsequently, we utilize the CLIP feature corresponding to the middle frame within each shot as the visual representation for that particular shot. The models then process these visual features as input and generate shot-level importance scores. These scores are instrumental in determining the selection of shots to be included in the trailer.

B More details about CMTD

B.1 Turning points annotation

Following previous studies [35, 36], we define turning points and list them in Table 5. The narrative structure of movies typically

comprises six key stages [17]: the setup, the new situation, progress, complications and higher stakes, the final push, and the aftermath, with five turning points dividing these stages. A turning point represents a critical juncture in the storyline, signaling a significant shift in the plot’s direction [44]. Our annotation of turning points is derived from the movie’s synopsis and includes both the chosen description of these turning points and their corresponding timestamps in the movie.

C More Quantitative Results

C.1 Robustness of hyperparameters

We further explore the impact of two key hyperparameters within our IPOT framework: (i) the weight δ of the BCE regularizer and (ii) the weight λ of the Entropy regularizer. We present the F1@K scores ($K = 1, 3, 5$) for various hyperparameter settings in Figure 6, illustrating the robustness of our approach to these parameters. The δ indicates the significance of the BCE regularizer, which is used to penalize the training of the movie shot selector. Figure 6(a) shows that our method achieves the best performance consistently when $\delta = 1$. The λ plays an important role in the Sinkhorn algorithm, usually affecting the iterative algorithm’s convergence speed and the results’ accuracy. This weight is associated with the values in the distance matrix (or grounding matrix) used. Under the current movie-music distance matrix, our IPOT method performs best when $\lambda = 1$.

C.2 Pretraining on MVs

We collect a set of 800 videos from the YouTube8M dataset that are tagged as “music video”, with an average duration of 4 minutes. We process the music videos the same way we process the trailers. In this case, we train the model through the correspondence between music video shots and audio shots rather than the correspondence between movie shots and trailer audio shots. Table 6 presents a performance comparison of different training strategies. Training on the music video (MV) dataset can bring better results in shot selection than only training on movie-trailer dataset. The possible reason is that MVs are relatively trimmed, contain fewer redundant shots, and are richer in quantity. The training strategy that involves pre-training on the music video dataset followed by fine-tuning on the movie-trailer dataset proves to be the most effective, yielding the best performance.

C.3 Analysis of spoiler ratio

According to the screenwriting theory [17], the expected positions or proportions of the five turning points (TPs) are listed in Table 7. Utilizing these theoretical segmentations, we calculate the mean and standard deviation of the proportion of official trailer shots in each stage, segmented by TPs, for both the training and test sets as indicated in Table 8. The standard deviations are generally large, suggesting that the distribution ratio of each stage within trailers varies considerably. We speculate that different movie categories and montage techniques will result in different stage distributions. It is notable that the proportion of Stage 6 (e.g., the aftermath) is consistently minimal. To avoid spoilers, we heuristically restrict our trailer generation to the first 90% of the movie content.

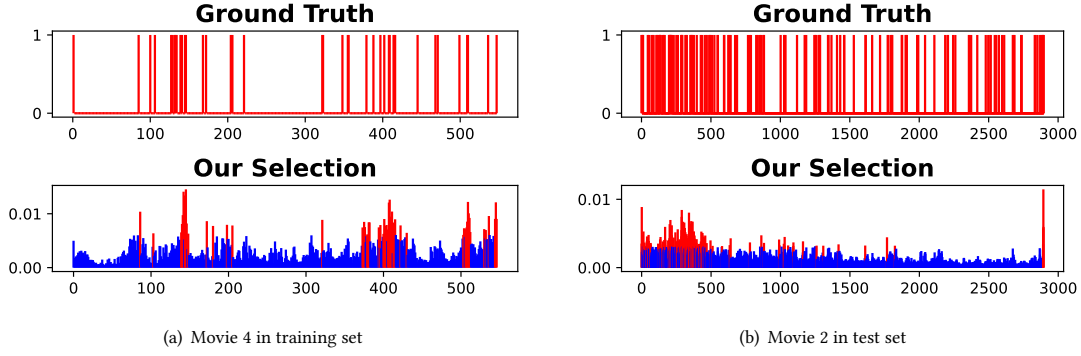


Figure 7: The comparison of the probability of each movie shot being selected as a trailer shot and the ground truth.

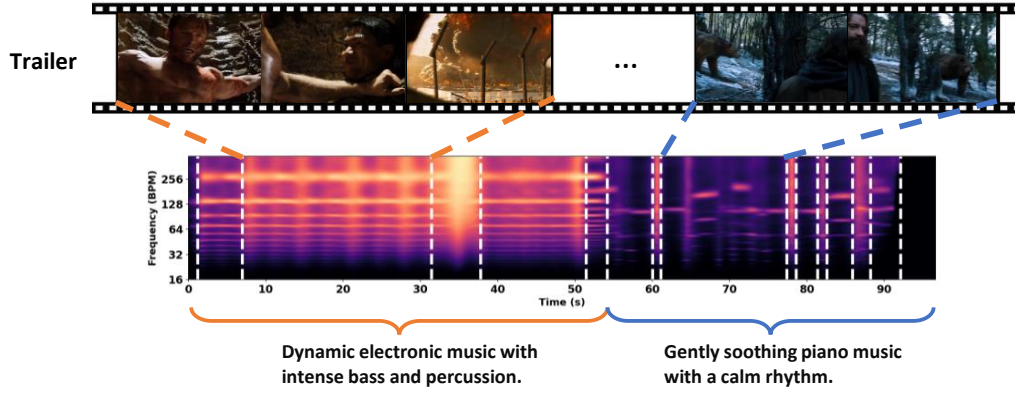


Figure 8: Comparison of movie shots selected by the model based on two different styles of music.

| | TP1 | TP2 | TP3 | TP4 | TP5 |
|--------|------|------|------|------|------|
| Theory | 10.0 | 25.0 | 50.0 | 75.0 | 94.5 |

Table 7: Expected TP position based on screenwriting theory.

| | | S1 | S2 | S3 | S4 | S5 | S6 |
|-------|------|-------|-------|-------|-------|-------|-------|
| Train | mean | 0.169 | 0.190 | 0.260 | 0.198 | 0.133 | 0.049 |
| | std | 0.131 | 0.125 | 0.127 | 0.110 | 0.093 | 0.067 |
| Test | mean | 0.237 | 0.142 | 0.247 | 0.221 | 0.125 | 0.028 |
| | std | 0.082 | 0.054 | 0.076 | 0.047 | 0.066 | 0.030 |

Table 8: The mean and standard deviation of the proportion of official trailer shots in each stage divided by TPs in the training set and test set, respectively.

D Visualization

Figure 7 shows the chosen shots' distribution based on probabilities output by our selector. The red lines in the ground truth indicate the movie shots selected into the official trailer, while the red lines in our selection indicate the trailer shots chosen by our selector. In

Figure 8, we concatenate two pieces of music with completely different rhythmic styles and then visualize them into a spectrogram. Lighter areas in the spectrogram represent higher energy levels, which usually correspond to louder sounds or larger amplitudes, implying a more exciting style of music, such as the piece included in the orange curly brace. Likewise, darker areas represent lower energy levels, often corresponding to more peaceful and soothing music, such as the piece included in the blue curly brace. Our model shows differences in the selection of movie shots for audio shots of different styles. For more energetic audio shots, it tends to select fight and explosion movie shots, while for smoother music, it tends to choose shots of walking or showcasing the surrounding environment. Figure 9 displays the comparison between generated trailers and the official trailer of the movie "The Wolf of Wall Street". Different methods generate their trailers using the audio from the official trailer. This visualization indicates that our conditional movie shot selector selects trailer shots more correctly than other methods.

E Future Work

Currently, the generated trailers are still incomparable to the human-edited trailers in quality, as shown in results of user study, which are far from practical applications. In the future, we plan to further

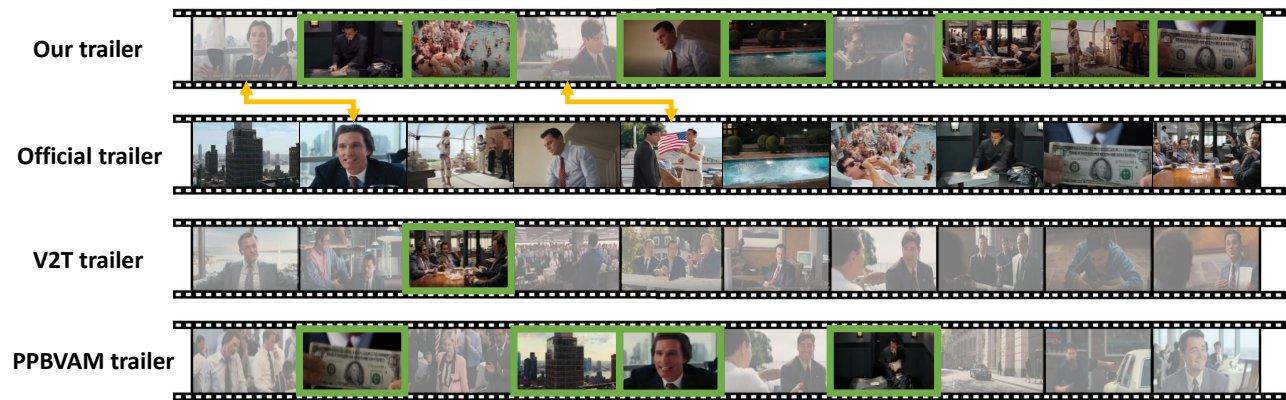


Figure 9: Comparison between some generated trailer shots and the official trailer shots of the movie "The Wolf of Wall Street" based on their appearance order. For each generated trailer, their correctly selected shots are marked with green boxes. The selected shot of our trailer connected to the shot in the official trailer by a yellow arrow means that they belong to the same scene.

enlarge out CMTD dataset, collecting more movies and trailers with metadata to support the learning of the model. In addition,

we would like to utilize more side information to learn the model, including but not limited to subtitles, turning points, and synopsis.