SSVMR: SALIENCY-BASED SELF-TRAINING FOR VIDEO-MUSIC RETRIEVAL

Xuxin Cheng, Zhihong Zhu, Hongxiang Li, Yaowei Li, Yuexian Zou*

ADSPLAB, School of ECE, Peking University, Shenzhen, China

ABSTRACT

With the rise of short videos, the demand for selecting appropriate background music (BGM) for a video has increased significantly. Video-music retrieval (VMR) task gradually draws much attention by research community. As other cross-modal learning tasks, existing VMR approaches usually attempt to measure the similarity between the video and music in the feature space. However, they (1) neglect the inevitable label noise; (2) neglect to enhance the ability to capture critical video clips. In this paper, we propose a novel saliency-based self-training framework, which is termed SSVMR. Specifically, we first explore to fully make use of the information containing in the training dataset by using a semi-supervised method to suppress the adverse impact of label noise problem, where a self-training approach is adopted. In addition, we propose to capture the saliency of video by mixing the two videos at span level and preserving the locality of the two original videos. Inspired by back translation in NLP, we also conduct back retrieval to obtain more training data. Experimental results on MVD dataset show that our SSVMR achieves the state-of-the-art performance by a large margin.

Index Terms— Video-music Retrieval, Cross-modal Matching, Self-training, Label Noise

1. INTRODUCTION

Background music (BGM) can adjust atmosphere, enhance emotional expression of the video, and achieve an immersive feeling for the audience. With the rise of short videos, the demand of selecting a proper BGM for a video has increased significantly. However, accomplishing this work manually requires a great deal of expertise. To handle this, video-music retrieval (VMR) has received increasing attention from researchers. As a cross-modal task, VMR aims to match the input video with the appropriate BGM from the audio library.

Recently, existing approaches attempt to measure the similarity between the video and music in the feature space. Some conventional VMR methods conduct cross-modal matching leveraging metadata [1][2][3][4][5]. Metadata typically provides key information about the content of videos or music in a quick and simple way, which makes multimedia content as searchable as text. [1] uses color histogram representations of album covers to index music pieces. [2] retrieves BGMs for videos by measuring similarity of their emotion tags. [5] leverages their emotion tags as joint constraints to pre-train music and video feature extractors, which can better align the embeddings of video and music modalities. Unfortunately, metadata is not always available, especially when the size of data is large.

More recently, some work has attempted to accomplish VMR tasks using a content-based approach without the use of metadata. [6] introduces a two-branch deep network that associates videos and music considering inter and intra-modal relationships. [7] proposes a cross-generation strategy which can also better align latent embedding of videos and music.

Despite the promising progress that existing VMR models have achieved, we discover they suffer from two main issues: (1) Neglecting the inevitable label noise in VMR [8]. Label noise can easily result in overfitting of the model [9]. Compared to other cross-modal counterparts, VMR task is likely to have more label noise for two reasons: a) The labels of VMR dataset are subjective, and different experts may choose very different BGMs for the same video; b) VMR dataset HIMV-50K [10] and MVD [11] both consist of three kinds of video-music pair, including official music videos, parody music videos and user-generated videos, resulting in some variation in the relationship between data and labels. (2) Neglecting to enhance the ability to capture critical video clips. Intuitively, different clips from the video play different roles in determining the corresponding music. There may be a small segment of the video that plays the most important role. However, previous work treats every video clip equally, which may lead to a decrease in model performance.

In this paper, we propose Saliency-based Self-training for Video-Music Retrieval (SSVMR) to tackle the above two issues. For the first issue, we use Gaussian Mixture Model (GMM) to select the samples which are more likely to be wrong and replace their original labels by soft labels through sharpen. Then, we introduce R-Drop [12] loss which regularizes two predicted distributions obtained by inputting the same sample twice to reduce the confirmation bias in self-training. For the second issue, we mix the two videos at span level relying on saliency. By doing so, more important information related to the prediction will be retained and the locality of the two original videos will be preserved. More-
over, inspired by back translation\cite{13} in NLP, we conduct back retrieval from music-video retrieval model to enlarge the dataset. Following \cite{10}, we train our model on HIMV-50K \cite{10} dataset and evaluate it on MVD\cite{11} dataset. Experiments and further analysis show that SSVMR can effectively handle label noise and improve the ability of model to capture critical video clips, which achieves performance over previous state-of-the-art methods by a large margin. To best of our knowledge, our framework is the first to handle label noise and consider saliency for VMR task.

2. METHOD

In this section, we first introduce the problem formulation of VMR in Section 2.1 and the backbone in Section 2.2, then describe the details of our framework. As shown in Fig.1, it mainly consists of self-training with R-Drop module in Section 2.3, mix at span level relying on saliency module in Section 2.4 and back retrieval module in Section 2.5.

2.1. Problem Formulation

The dataset for VMR task usually contains video-music pairs, which can be denoted as \( D = \{(v, m)\} \), where \( v \) is the embedded features of the video, and \( m \) is the corresponding embedded features of the BGM. For a set of video \( V \) and a set of music \( M \), a VMR method aims to retrieve a list of music \( m \) in \( C(m) \) is a potential match for the query video \( v \) from \( V \).

2.2. Backbone

We use CBVMR\cite{3} as the backbone, which uses a two-branch neural network and combines the inter-modal ranking constraint with soft intra-modal structure constraint to construct an unified embedding network. It tries to minimize the empirical risk by applying the triplet loss \( L_T \) as follows:

\[
L_T = \sum_i \ell_i = \lambda_1 \sum_{i \neq j} \max (0, v_i^\top m_j - v_i^\top m_i + \epsilon) \\
+ \lambda_2 \sum_{i \neq j} \max (0, m_i^\top v_j - m_i^\top v_i + \epsilon) \\
+ \lambda_3 \sum_{i \neq j \neq k} C_{ijk}(v) (v_i^\top v_j - v_i^\top v_k) \\
+ \lambda_4 \sum_{i \neq j \neq k} C_{ijk}(m) (m_i^\top m_j - m_i^\top m_k)
\]

(1)

Here, \( \lambda_1 \) and \( \lambda_2 \) balance the impact of inter-modal ranking loss from video-to-music and music-to-video matching. And the impact of soft intra-modal structure loss within each modality is balanced by \( \lambda_3 \) and \( \lambda_4 \), respectively. \( \epsilon \) is the margin for triplet loss and function \( C(\cdot) \) is defined as follows:

\[
C_{ijk}(x) = \text{sign} (x_i^\top x_k - x_i^\top x_j) - \text{sign} (\bar{x}_i^\top \bar{x}_k - \bar{x}_i^\top \bar{x}_j)
\]

(2)

where \( x_i, x_j, \) and \( x_k \) are the trainable features in multimodal space and \( \bar{x}_i, \bar{x}_j, \) and \( \bar{x}_k \) are intra-modal features before passing through the embedding network. We first warm up the model using \( L_T \) to enable it to perform the preliminary VMR task without overfitting the noisy labels and then perform SSVMR for the rest epochs.

2.3. Semi-Supervised Self-Training using R-Drop

As mentioned in \cite{14}, noisy samples usually have higher losses in the early stage. During the training process, the loss distributions of clean and noisy samples tend to obey two Gaussian distributions\cite{15}, where the loss of noise samples holds a bigger mean value. Therefore, we apply the widely used GMM to distinguish noisy samples by the loss of each input sample. We use GMM which has 2 components and
fit it to the observations using the Expectation-Maximization (EM) algorithm. To be specific, we first calculate the probability $w_i$ that each sample belongs to the Gaussian component with bigger mean $g$, i.e., the probability of belonging to the noisy sample, as follows:

$$w_i = p(g \mid \ell_i)$$  \hspace{1cm} (3)

Then, we set a threshold $\tau$ for the probability. The original dataset $D$ is divided into two parts, including a labeled set $X$ and an unlabeled set $U$. The label of the sample in unlabeled set $U$ is most likely wrong, so we replace it by a new label. We generate the new soft label $\hat{m}$ by sharpening the predicted distribution of the model to convert unlabeled set $U$ to labeled set $\hat{U}$, which can encourage the model to produce lower-entropy predictions:

$$X = \{(v_i, m_i) \mid v_i \in D, w_i \leq \tau\}$$  \hspace{1cm} (4)

$$U = \{(v_i) \mid v_i \in D, w_i > \tau\}$$  \hspace{1cm} (5)

$$\hat{m} = \text{Sharpen}(p(v, \theta), T)$$  \hspace{1cm} (6)

$$\hat{U} = \{(v_i, \hat{m}_i) \mid v_i \in U\}$$  \hspace{1cm} (7)

$$\hat{D} = X \cup \hat{U}$$  \hspace{1cm} (8)

where $p(v, \theta)$ denotes the prediction of sample $v$, $\text{Sharpen}(\cdot)$ is the temperature sharpening function commonly used in self-training and $T$ denotes the temperature:

$$\text{Sharpen}(p, T) = p_i^\frac{1}{T} / \sum_{j=1}^{L} p_j^\frac{1}{T}$$  \hspace{1cm} (9)

There is an inconsistency between training and inference of the dropout based models, because dropout is used in training but not in inference, which will be more evident in noisy samples and cause more harm to the model. So we introduced R-Drop[12] loss $L_R$ to overcome the inconsistency:

$$L_R = \sum_{m \in U} \frac{1}{2} \left( \mathcal{D}_{KL}(p_1(m, \theta) \| p_2(m, \theta)) + \mathcal{D}_{KL}(p_2(m, \theta) \| p_1(m, \theta)) \right)$$  \hspace{1cm} (10)

where $p_1(m, \theta)$ and $p_2(m, \theta)$ are two predicted distributions obtained by inputting the same sample twice. $\mathcal{D}_{KL}(a\|b)$ denotes Kullback-Leibler divergence, which plays a role in regularizing these two probability distributions. We combine the clean samples with original labels and the noisy samples with generated soft labels to obtain the new dataset $\hat{D}$.

2.4. Mix at Span Level Relying on Saliency

There may be a small segment of the video that plays the most critical role in the outcome of the retrieval music. Inspired by [16], to improve the ability of our framework to capture critical video information, we mix the two videos at span level. First, we compute the gradient of $L_T$ with respect to input embedding $v$, and use its magnitude as the saliency:

$$s = \| \partial L \|_2$$  \hspace{1cm} (11)

where $\| \cdot \|_2$ denotes L2 norm. For two video-music pairs denoted as $(v_1, m_1)$ and $(v_2, m_2)$, we select the least salient span $x_{2}^S$ from $v_2$ and replace it with the most salient span $x_{1}^S$ from $v_2$ to obtain a new video embedding $\hat{v}$ which is the concatenation of $(v_1^L; v_2^S; v_1^R)$ where $v_1^L$ and $v_1^R$ are embedding located to the left and the right side of $x_{1}^S$ respectively in the original video $x_1$. The selected span length of these two videos are the same and are calculated as follows:

$$l_1 = l_2 = \max \left( \min \left( \frac{v_0}{v_1}, \frac{v_2}{v_3} \right), 1 \right)$$  \hspace{1cm} (12)

where $l_1$ and $l_2$ are the length and $\lambda_0$ is the initial mixup ratio. We calculate the triplet loss $L_T$ of the augmented output with original video embedding of each sample and combine them by weight $\lambda$ to obtain the mix loss $L_M$:

$$L_M = \lambda L_T(\hat{v}, m_1) + (1 - \lambda) L_T(\hat{v}, m_2)$$  \hspace{1cm} (13)

With the triplet loss and mix loss, the final training objective is as follows:

$$L = L_T + L_R + L_M$$  \hspace{1cm} (15)

2.5. Back Retrieval

Before training video-music retrieval model as above, we conduct back retrieval to obtain more data. In the neural machine translation task, back translation[13] is widely used to augment the parallel training corpus. Inspired by this, we first train a music-video retrieval model with the original dataset, and then feed each music into the model to obtain the three most similar videos. Considering that the matching criteria between video and music is more ambiguous than that of other cross-modal tasks such as image-to-text retrieval, we randomly select one of the three videos and combine it with the music into a new pair of data.

3. EXPERIMENTS

3.1. Datasets and Settings

Following [10], we train our model on HIMV-50K[10] dataset and evaluate it on MVD[11] dataset. HIMV-50K is a large collection of musical videos corresponding to the part of clips of the YouTube-8M[17] dataset annotated as “music video”, which consists of 50,000 video-music pairs without annotations. Note that while the audio of these clips always contains music, the video can be anything from professional promotional clips to amateur montages of still images, i.e., the quality of the video part of the clips is very heterogeneous. Therefore, we use MVD[11] dataset to evaluate our model. It consists of 2,212 music video clips of professional quality. The average duration of each clip is 4 minutes. We randomly selected 2,000 of these clips to evaluate our model.

In this paper, the parameters $\lambda_1$, $\lambda_2$, $\lambda_3$, $\lambda_4$ and $e$ are set as 3, 1, 0.2, 0.2 and 6 respectively. Temperature $T$ is set as 0.8, threshold $\tau$ is set as 0.3 and $\lambda_0$ is set as 0.4. We use Adam[18] to optimize the parameters in our model with the learning rate $4e\cdot4$ and use dropout with probability 0.9. In addition, we use Recall@K[19] to evaluate the model performance, which is a standard metric for cross-modal retrieval that calculates the
average hit ratio of the matched music clips over all the music clips that are ranked with top K match scores.

3.2. Main Results
Table 1 shows our experiment results on the MVD dataset and the performance of the reference VMR systems. We have the following observations: 1) Compared with baseline CMMR[5] and CBVMR[6], our proposed framework gain a large improvement. The reason is that SSVMR considers the label noise which is extremely harmful to training. It’s worth noticing that the parameters between SSVMR and CBVMR are of the same magnitude, which further verifies that the contribution of SSVMR comes from the semi-supervised self-training using R-Drop rather than parameters factor. 2) Although CMVAE[7] models the matching of relevant music to video as a multimodal cross-generation problem with product-of-experts principle as the fusion strategy, SSVMR still outperforms it. We attribute it to the fact that SSVMR can capture critical video information and CMVAE is proposed to retrieve music for video on the dataset, where a music clip could be associated to multiple videos. However, MVD is a dataset that associates one music clip to one video only, which can’t provide sufficient information for CMVAE to train an encoder network well. 3) MRCMV can efficiently improve performance by leveraging the widely added voice-overs, which are the voices of unseen narrators. But when voice-overs are not added, SSVMR outperforms it, which verifies that our framework can leverage original data better.

<table>
<thead>
<tr>
<th>Models</th>
<th>R@1</th>
<th>R@10</th>
<th>R@25</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSVMR</td>
<td>4.45</td>
<td>15.20</td>
<td>24.50</td>
</tr>
<tr>
<td>Improve</td>
<td>34.8%</td>
<td>20.2%</td>
<td>34.6%</td>
</tr>
</tbody>
</table>

Table 1: Results on MVD datasets. Best results are in bold.

3.3. Analysis
3.3.1. Ablation Test
To study the effectiveness of each component in our approach, we perform ablation experiments on the MVD dataset, whose results are shown in Table 2.

We remove the back retrieval module, which is called w/o BR in Table 2. From the results, we can observe that the absence of back retrieval leads to 0.55% R@1 drop, which indicates that back retrieval enlarges the dataset and encourages the robustness of our model.

We further remove the mix relying on saliency module, named as w/o BR + Mix in Table 2. We can obviously observe the R@1 drops by 1.30%. This indicates that mixing the two videos at span level, which retains more information related to the prediction relying on saliency, can improve the ability of model to capture critical video information so that the model can retrieval a music better.

We finally remove the R-Drop module and then remove the self-training module, denoted as w/o BR + Mix + R-Drop and w/o BR + Mix + R-Drop + SL respectively. The R@1 also drops, which verifies that these modules help to handle the label noise and are beneficial for the model.

3.3.2. Effect of the number of spans replaced
When we mix two videos at span level, only one span of the video is replaced. To check that if it’s enough to only choose one span, we conduct a series of experiments shown in Table 3.

We observe that as the number of span increases, R@1 drops gradually. The reason is that replacing too many spans will break semantics and be harmful to model instead.

<table>
<thead>
<tr>
<th>N</th>
<th>R@1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4.45</td>
</tr>
<tr>
<td>2</td>
<td>4.27(↓0.18)</td>
</tr>
<tr>
<td>3</td>
<td>4.03(↓0.42)</td>
</tr>
<tr>
<td>4</td>
<td>3.87(↓0.58)</td>
</tr>
</tbody>
</table>

Table 2: Ablation results on MVD.

Table 3: Results on MVD with different number of span.

4. CONCLUSION
In this paper, we propose Saliency-based Self-training for Video-Music Retrieval (SSVMR) to use semi-supervised self-training with R-Drop. Besides, we mix the two videos at span level relying saliency and use back retrieval to enlarge the dataset. Experiments and analysis demonstrate the effectiveness of our proposed method, which can handle the label noise, enhance the ability to capture critical video information and enhance the robustness. In the future, we will explore how to add pre-training techniques to our approach to further improve the performance of the model.

5. ACKNOWLEDGEMENTS
This paper was partially supported by Shenzhen Science & Technology Research Program (No: GXWD20201231165807007-20200814115301001) and NSFC (No: 62176008).
6. REFERENCES


