A Multimodal Framework for Video Ads Understanding

Zejia Weng, Lingchen Meng, Rui Wang, Zuxuan Wu, Yu-Gang Jiang
Shanghai Key Lab of Intelligent Information Processing,
School of Computer Science, Fudan University, Shanghai, China
{zjweng20,lcmeng20,ruiwang16,zxwu,ygj}@fudan.edu.cn

ABSTRACT

There is a growing trend in placing video advertisements on social platforms for online marketing, which demands automatic approaches to understand the contents of advertisements effectively. Taking the 2021 TAAC competition as an opportunity, we developed a multimodal system to improve the ability of structured analysis of advertising video content. In our framework, we break down the video structuring analysis problem into two tasks, *i.e.*, scene segmentation and multi-modal tagging. In scene segmentation, we build upon a temporal convolution module for temporal modeling to predict whether adjacent frames belong to the same scene. In multi-modal tagging, we first compute clip-level visual features by aggregating frame-level features with NeXt-SoftDBoF. The visual features are further complemented with textual features that are derived using a global-local attention mechanism to extract useful information from OCR (Optical Character Recognition) and ASR (Audio Speech Recognition) outputs. Our solution achieved a score of 0.2470 measured in consideration of localization and prediction accuracy, ranking fourth in the 2021 TAAC final leaderboard.

CCS CONCEPTS

• **Computing methodologies** → **Scene understanding**; • **Information systems** → **Multimedia and multimodal retrieval**.

KEYWORDS

Scene Segmentation, Multi-Modal Tagging, Global-Local Attention, Temporal Modeling

ACM Reference Format:
Zejia Weng, Lingchen Meng, Rui Wang, Zuxuan Wu, Yu-Gang Jiang. 2021. A Multimodal Framework for Video Ads Understanding . In *Proceedings of the 29th ACM International Conference on Multimedia (MM '21), October 20–24, 2021, Virtual Event, China.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3474085.3479202

1 INTRODUCTION

Online video advertising is an effective marketing method due to its advantages such as strong flexibility, widespread, low cost, and strong interaction. Therefore, different companies have invested more and more effort to produce online video advertisements, and use different social platforms to accurately deliver them to users. Due to the important value of video advertising, there are also many studies related to video advertising, including advertisement recommendation, quality monitoring, interruption time estimation, etc. With the rapid development of the 5G field, the number of video advertisements has also increased rapidly, and thus it is critical to understand the structures of video ads automatically and effectively.

In contrast to traditional video ads classification, video ads structure understanding requires the model to be able to segment the advertisements into different scenes correctly, and then perform multi-label classification for each scene. Therefore, in essence, video ads content structuring can be divided into two relatively independent tasks, which are scene segmentation and multi-label classification.

How to effectively identify scene boundaries remains a challenging problem for scene segmentation. One method is to divide the task into two stages, shot detection and shot merging, as it is beneficial to reduce the number of potential turning points to be judged [1] . However, the boundary error caused by shot segmentation will affect the scene segmentation quality, and no shot boundary information is given on the 2021 TAAC data set, making it difficult to develop a convincing shot segmentation method. Therefore, we use an end-to-end scene segmentation approach, directly predicting whether each time position is a scene turning point by observing its surrounding video frames.

Videos are multi-modal in nature, which has motivated extensive work to leverage different modalities like visual and audio for better video understanding [2–5]. Compared to visual and audio information, textual clues are less explored for video understanding. However, in video ads structuring understanding, textual information plays an important role since the dialogue information and subtitles in the ads are oftentimes directly related to content topics, such as product categories, names of the promotional items, types of advertisements, and *etc.* To this end, in the 2021 TAAC Challenge, we focus on OCR and ASR features in addition to visual modalities, and propose global-local attention to combine OCR and ASR features to fully exploit textual information for improving the overall performance.

2 APPROACH

Given a test video, our goal is to learn how to correctly segment video scenes and effectively make multi-label predictions for each scene, so as to analyze video ads at a fine-grained level. We achieve this with a two-stage multi-modal analysis framework, which operates on different modalities, including audio dialogues, appearance clues, and textual information among key frames. Figure 1 presents an overview of the framework. Below, we first introduce the preliminaries of our work, and then elaborate on how we solve the video ads content structuring problem in two stages, which are scene segmentation and multi-modal tagging.

MM '21, October 20–24, 2021, Virtual Event, China

ACM ISBN 978-1-4503-8651-7/21/10…$15.00
https://doi.org/10.1145/3474085.3479202

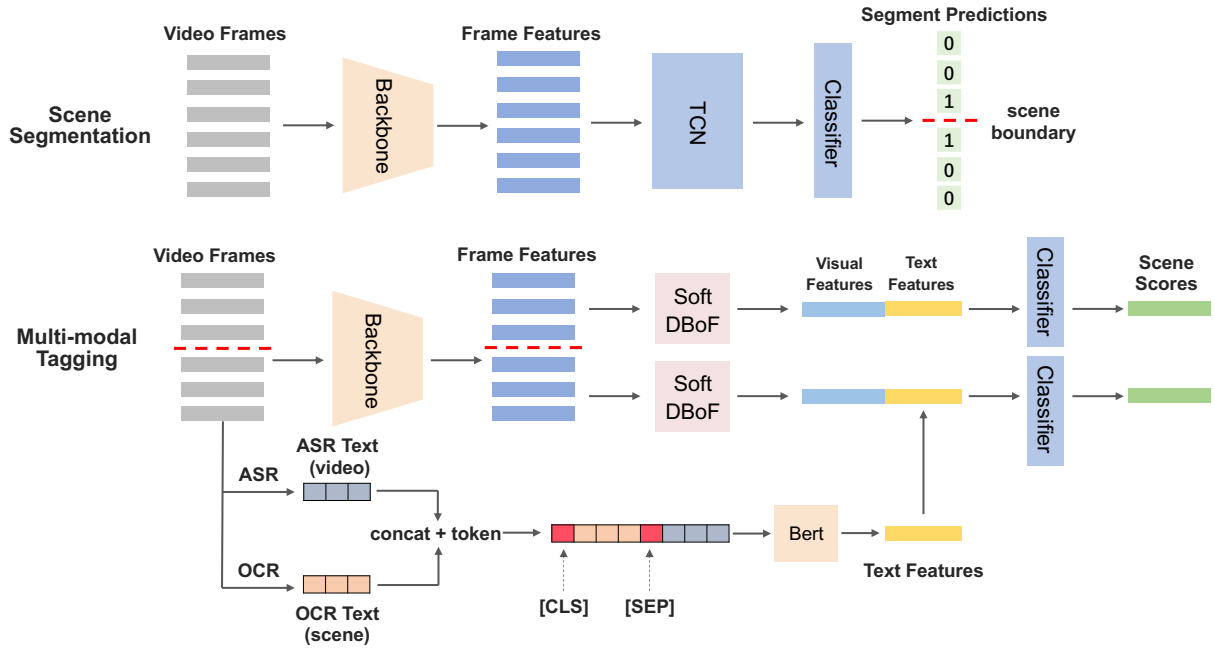

Figure 1: An overview of our framework, which consists of scene segmentation and multi-modal tagging.

2.1 Preliminaries

Given a training set with N video clips, where each video is associated with two types of modalities, *i.e.*, audio and appearance. Thus, we can represent a video with T video snippets as:

$$V = [V_1, V_2, V_t, \cdots, V_T] \tag{1}$$

$$\text{where } V_t = \{V_t^A, V_t^I\}. \tag{2}$$

Here, V_t contains V_t^A *and* V_t^I denote the audio and image modality. Below we introduce how to extract useful information from the two modalities.

Audio Modality. Audio signals often contain important contextual information and are helpful to improve the accuracy of video classification. Because the audio information in video ads mainly exists in the form of dialogues, we use ASR to convert the original audio signals into texts, and then perform recognition based on recognized text from dialogues. However, extracting local ASR information in a short time window is challenging as short video clips tend to contain discontinuous speech and confusing context. While global ASR outputs for the entire video are provided by the organizers, they lack temporal annotations at the scene-level and it is challenging to perform grounding to associate texts with scenes. Since we need ASR clues for fine-grained understanding, we will introduce a global-local attention mechanism to mitigate this issue. Formally, we define the mapping from audio signals of a whole video to a speech record as $f_{\theta_A} : V^A \mapsto \boldsymbol{x}^A$, where f_{θ_A} denotes the weights of the audio speech recognition network and $\boldsymbol{x}^A$ is the speech.

Image Modality. We use a 2D CNN model to capture appearance information from RGB frames. Compared to audio clues which are usually noisy, the appearance network provides decent visual information with a moderate computational cost and it suffices in most cases. To this end, we respectively use ResNet101, EfficientNet B4 and EfficientNet B5 [6] backbone to extract appearance clues. Also, we adopt a 3D CNN model to capture motion clues that depict how objects move among stacked frames. We instantiate the motion network with a SlowFast network [7]. Formally, we define the mapping from a stack of frames or an RGB frame to a feature vector as $f_{\theta_I} : V_t^I \mapsto \boldsymbol{x}_t^I$, where f_{θ_I} is the weights of the network (ResNet101, EfficentNet B4, EfficientNet B5 or SlowFast) and $\boldsymbol{x}_t^I$ represents visual features. In addition, the 2D CNN models are pretrained on ImageNet and the 3D CNN model are pretrained on Kinetics-400, and then all the models are finetuned on the ads dataset. More details can be found in section 3.1.3.

Textual Information. Optical characters in videos, such as subtitles and signboards, often provide rich information to improve video classification accuracy. We apply OC-OCR [8] to capture optical character information for each frame. Formally, we define the mapping from a frame to a set of character sequences as $f_{\theta_T} : V^I \mapsto \boldsymbol{X}^T$, where f_{θ_T} is the weights of the OCR model, V^I is a frame in video and $\boldsymbol{X}^T = \{\boldsymbol{x}_1^T, \boldsymbol{x}_2^T, ..., \boldsymbol{x}_N^T\}$ is a set of recognition character sequence in the frame. Besides, optical character information between adjacent frames is often redundant. To overcome this problem, we propose a post-processing method to remove the same text in adjacent frames.

2.2 Temporal Convolution Networks for Scene Segmentation

We build upon Temporal Convolution Network (TCN) due to its effectiveness for action segmentation. Briefly, we firstly stack two TCN blocks as the shallow layers of the network, and each TCN block consists of four 1-D dilated convolutional layers, whose dilated sizes are 1, 2, 4, 8 respectively. Then, intuitively, we use a normal 1-d convolution in the last layer of the network, to avoid confusing boundaries. We also use dropout to prevent overfitting. We consider the scene segmentation problem as a binary classification problem at each time step. Therefore, after we obtain the new representation of each time step by multiple TCNs, we will classify it through fully connected layers. Since ground-truth provided by the organizers only contain one transition point for each scene, we argue that such annotations will lead to biased training sets. We modify the original ground-truth labels by setting both the beginning and the end of scenes as transition points (See Fig. 2 for an illustration). Then we optimize the TCN network using a standard binary cross-entropy loss. We further use focal loss [9] for improved performance.

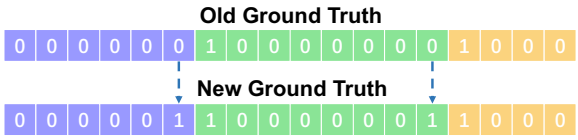

Figure 2: We modify the original GT by setting both the beginning and the end of scenes as transition points.

2.3 Multi-Modal Tagging for Scenes

Video ads are multi-modal in nature. In 2021 TAAC, we use image and textual information for multi-label video classification. For visual information, we introduce a NeXt-SoftDBoF module for temporal aggregation. Compared with NeXtVLAD [10], NeXt-SoftDBoF has fewer parameters and is less prone to overfitting. For text sequences, we design a global-local attention mechanism to combine ASR and OCR outputs. Finally, we concatenate the logits values from two modalities for improved performance.

2.3.1 NeXt-SoftDBoF.

We build upon NeXtVLAD to construct our NeXt-SoftDBoF Aggregation network, which can be regarded as a variant of SoftBoF(soft bag of features)[11]. The core idea is to expand the original K cluster centers to $G * K$ and guarantee the final word frequency statistics dimension is still K with a multi-head operation. Specifically, the input features firstly complete the soft-word frequency statistics at the K centers in each group, and then an attention module will assign different weights to the G groups to achieve weighted summation.

We firstly expand the input vector x_i as $\dot{x}_i$ with a dimension of λN via a linear fully-connected layer, where λ is the expansion multiple and N is the dimension of x_i. Then NeXt-SoftDBoF is formalized as follows,

$$y_k = \sum_{\substack{i\in\{1,\ldots,T\}\\ g\in\{1,\ldots,G\}}} \alpha_g(\dot{x}_i)\alpha_{gk}(\dot{x}_i) \tag{3}$$

$$\alpha_g(\dot{x}_i) = \sigma(\hat{w}_g^T \dot{x}_i + \hat{b}_g) \tag{4}$$

$$\alpha_{gk}(\dot{x}_i) = \frac{e^{w_{gk}^T \dot{x}_i + b_{gk}}}{\sum_{s=1}^{K} e^{w_{gs}^T \dot{x}_i + b_{gs}}} \tag{5}$$

where $\alpha_g(\dot{x}_i)$ represents the attention weights of the g-th group and $\alpha_{gk}(\dot{x}_i)$ represents the "frequency" of the i-th feature belonging to the k-th cluster center in the g-th group. Finally, we get the result of aggregation as $\boldsymbol{y} = [y1, y2, ..., y_k]$.

Table 1: Comparisons between OCR and ASR.

text type	pros	cons
OCR	has temporal locality	Noisy
ASR	Concise and accurate	Global

2.3.2 Global-Local Attention. Texts in ads provide rich information. As mentioned in Section 2.1, we have obtained two kinds of textual information: OCR in each frame and ASR for the entire video. The characters of OCR and ASR are summarized in Table 1. Texts from OCR are usually noisy or redundant. for example, in the ads of math homework guidance, OCR may recognize a bunch of math formulas, resulting in very noisy OCR outputs. ASR mainly records the dialogue of the characters, which is always relevant to the video content and is more concise and accurate than the OCR text. However, ASR text lacks temporal locality.

In order to solve these problems, we introduce a global-local attention mechanism, hoping that the global ASR and local OCR information can pay attention to each other and learn from each other. We concatenate the OCR text and the ASR text directly and use the <SEP> token to separate them. More specifically, we follow the form of "<CLS>[OCR text]<SEP>[ASR text]" to construct the input text for the BERT model [12].

For each scene, our model predicts the confidence scores of C classes, where $C = 82$ in TAAC 2021. Since scene classification is a multi-label task in TAAC 2021, we calculate BCE loss for each class, then sum them up to get the final tagging loss.

3 EXPERIMENTS

3.1 Experimental Setup.

3.1.1 Dataset. The 2021 TAAC video ads dataset contains 5,000 training videos and 5000 testing videos, which are collected from online video advertisements. For each video, the boundary and the labels of each scene are provided as ground-truth annotations. To evaluate our method, we randomly sample 10% videos from the training set as the local validation set and utilize the other data for training.

3.1.2 Evaluation. $F1@0.5s$ is used for the evaluation of scene segmentation. Specifically, the predicted boundary of a scene segment

Table 2: Performance. SF: SlowFast; E4: EfficientNet-B4; E5: EfficientNet-B5; R: ResNet101. "n* " indicates models produced by different seeds. If GL-Attention is ✗, only OCR features will be used, else, global(ASR)-local(OCR) attention will be added.

Exp Name	SceneSeg			Multi-Modal Tagging			threshold	F1*mAP@20
	Backbone	Finetune	Focal Loss	Backbone	Finetune	GL-Atttention		
Baseline	R	✗	✗	R	✗	✗	0.5	0.1565
I	R	✗	✗	R	✓	✗	0.5	0.1894
II	R	✓	✗	R	✓	✗		0.1964
III	R	✓	✗	SF	✓	✗		0.2051
IV	R	✓	✗	R+SF	✓	✗		0.2122
V	R	✓	✓	R+SF	✓	✗		0.2141
VI	R	✓	✗	R+SF	✓	✓		0.2230
VII	R+E4	✓	✗	R+SF+E4	✓	✓		0.2327
VIII	3*(R+E4+E5)	✓	✗	1*R+2*E4+3*SF	✓	✓	0.5	0.2389
IX	3*(R+E4+E5)	✓	✗	1*R+2*E4+3*SF	✓	✓	0.45	0.2419
X	4*(R+E4+E5)	✓	✗	3*R+3*E4+6*SF	✓	✓	0.45	0.2438
XI	4*(R+E4+E5)	✓	✗	3*R+3*E4+6*SF	✓	✓	0.4	**0.2455**
XII	4*(R+E4+E5)	✓	✗	3*R+3*E4+6*SF	✓	✓	0.35	**0.2470**

is true positive if the gap between the ground-truth and the prediction is less than 0.5s; otherwise, it is a false positive. Each ground truth only matches a prediction once.

The predictions of scene tagging are evaluated by the mean Average Precision (mAP) with overlapping thresholds between 0.5 and 0.95 (stride 0.05), denoted by *average mAP@IoU* = 0.5 : 0.05 : 0.95. The overlapping threshold is determined based on the temporal intersection over union (tIoU) ratio between predicted segments and ground-truth. For the evaluation of both the scene segmentation and the scene tagging, the final results are computed by the multiplication of F1-score and average mAP.

3.1.3 Implementation details. We train our models on Tesla V100 GPUs using PyTorch 1.7. We use the Adam optimizer [13] and the exponential moving average (EMA) with the weight of 0.9 for all training stages. When training models for scene segmentation, the learning rate is set as $1e^{-4}$ with a batch size of 64. When training models for scene classification, the learning rate is set to $1e^{-4}$ for the classifier and $1e^{-5}$ for other modules, and the batch size is 32.

3.2 Main Results and Discussion

The comparison results are summarized in Table 2.

Baseline : We use ResNet101 pretrained on ImageNet to capture frame features and it offers a score of 0.1565.

Exp I : We fine-tune the ResNet101 on TAAC 2021 scene classification and use its features for multi-label scene classification. The score is 0.1894, which achieves a significant increase of 0.0329 compared with the baseline. These results validate our hypothesis that fine-tuning model is beneficial to improve feature quality and scene classification accuracy.

Exp II : We utilize the fine-tuned ResNet101 features on the scene segmentation task, and also achieve higher performance. It indicates that fine-tuned features include better advertising semantics so that it can make better scene segmentation.

Exp III : In this experiment, we use fine-tuned SlowFast features for the scene classification task. It achieves a 0.087 score increase compared with fine-tuned ResNet101 features. Unlike 2D models, SlowFast captures more motion information. We have also tried 3D features, such as SlowFast and I3d, for scene segmentation. However, it doesn't work on our validation set. We assume that 3D models may not be suitable for scene segmentation since clip features are not temporally aligned.

Exp IV : We use ResNet101 features and SlowFast features for ensembling, which offers an 0.2122 score. It proves the effectiveness of model ensembling.

Exp V: We use focal loss when fine-tuning ResNet101 on the scene segmentation task. It only achieves a slight improvement. Limited by the number of competition submissions, we don't use it in our final solution.

Exp VI: In this experiment, we use ASR text and apply GL-Attention between ASR text and OCR text. It achieves a 0.0108 score increase, which proves ASR is effective for scene understanding.

Exp VII : Based on Exp VI, we use EfficientNet B4 for both two tasks, which achieves almost 1 % improvement. It indicates that ensemble is also useful for scene segmentation. Besides, ensembling more features can improve performance.

Exp VIII - XII: To boost the performance further, we set different seeds to diversify model initialization and the train/val split. Meanwhile, we study the influence of the threshold on the results and finally achieve a score of 0.2470.

4 CONCLUSION

In this paper, we introduce a multimodal framework for video ads structuring understanding, which can effectively segment and predict video scenes. The system leverages global ASR information, local OCR information and visual clues for improved performance. The system ranked fourth in the 2021 TAAC leaderboard, which proves its effectiveness.

REFERENCES

[1] Anyi Rao, Linning Xu, Yu Xiong, Guodong Xu, Qingqiu Huang, Bolei Zhou, and Dahua Lin. A local-to-global approach to multi-modal movie scene segmentation. In *CVPR*, 2020.

[2] I-Hong Jhuo, Guangnan Ye, Shenghua Gao, Dong Liu, Yu-Gang Jiang, D. T. Lee, and Shih-Fu Chang. Discovering joint audio-visual codewords for video event detection. *Mach Vis Appl*, 2014.

[3] Zuxuan Wu, Yu-Gang Jiang, Xi Wang, Hao Ye, and Xiangyang Xue. Multi-stream multi-class fusion of deep networks for video classification. In *ACM Multimedia*, 2016.

[4] Ruohan Gao, Tae-Hyun Oh, Kristen Grauman, and Lorenzo Torresani. Listen to look: Action recognition by previewing audio. In *CVPR*, 2020.

[5] Zejia Weng, Zuxuan Wu, Hengduo Li, and Yu-Gang Jiang. Hms: Hierarchical modality selection for efficient video recognition. *arXiv e-prints*, 2021.

[6] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, 2019.

[7] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019.

[8] Lingchen Meng. Oc-ocr. https://github.com/MengLcool/Oc-OCR, 2021.

[9] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017.

[10] Rongcheng Lin, Jing Xiao, and Jianping Fan. Nextvlad: An efficient neural network to aggregate frame-level features for large-scale video classification. In *ECCV Workshops*, 2018.

[11] Antoine Miech, Ivan Laptev, and Josef Sivic. Learnable pooling with context gating for video classification. *arXiv preprint arXiv:1706.06905*, 2017.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[13] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.