What You Look Matters? Offline Evaluation of Advertising Creatives for Cold-start Problem

Zhichen Zhao, Lei Li, Bowen Zhang, Meng Wang,Yuning Jiang, Li Xu, Fengkun Wang
Wei-Ying Ma
{zhaozhichen.water,lileilab}@bytedance.com
Bytedance Inc

ABSTRACT

Modern online-auction-based advertising systems utilize user and item features to automatically place ads. In order to train a model to rank the most profitable ads, new ad creatives have to be placed online for hours to receive sufficient user-click data. This corresponds to the cold-start stage. Random strategy lead to inefficiency and inferior selections of potential ads. In this paper, we analyze the effectiveness of content-based selection during the cold-start stage. Specifically, we propose Pre Evaluation of Ad Creative Model (PEAC), a novel method to evaluate and select ad creatives offline before being placed online. Our proposed PEAC utilizes the automatically extracted deep feature from ad content to predict and rank their potential online placement performance. It does not rely on any user-click data, which is scarce during the cold-starting phase. A large-scale system based on our method has been deployed in a real online advertising platform. The online A/B testing shows the ads system with PEAC pre-ranking obtains significant improvement in revenue gain compared to the prior system. Furthermore, we provide detailed analyses on what the model learned, which gives further suggestions to improve ad creative design.

KEYWORDS

Advertisement ranking; Deep Neural Networks; Cold start

ACM Reference Format:
Zhichen Zhao, Lei Li, Bowen Zhang, Meng Wang,Yuning Jiang, Li Xu, Fengkun Wang and Wei-Ying Ma. 2019. What You Look Matters? Offline Evaluation of Advertising Creatives for Cold-start Problem. In *The 28th ACM International Conference on Information and Knowledge Management (CIKM'19), November 3–7, 2019, Beijing, China.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3357384.3357813

1 INTRODUCTION

Online advertising has been the major business revenue source for many internet products, including Google's and Baidu's search engines, Facebook's social platform, ByteDance's and Kwai's information feeds. Modern online ad platforms display personalized ad creatives according to user's interest and likes. Once a user catches/clicks the ads, it will be considered as a potential buyer and the advertisers will be charged based on the impressions or actions. However, to keep satisfactory user experience, only limited number of slots will be available for ad showing. An ad ranking system selects the most valuable and high-quality ad creatives to display online. Specifically, CTR/CVR models are developed to predict CTRs/CVRs and rankings are determined by $CTR \times CVR \times bid$.

State-of-the-art ad ranking algorithms (e.g. CTR prediction) heavily rely on the ID features of the ads. Each ad is assigned a unique ad ID when it is created. Then an embedding corresponding to its ID is learned through historical information, such as historical clicks, user stay time and like/dislike feedback. Finally the embedding will be used as the ID feature to predict the CTR or CVR in future impressions. For each new ad, the procedure to learn a "matured" embedding from random initialization is called cold start process. Cold start problem is one of the most critical problems in current advertising systems since the ads with "unmatured" embedding will occupy plenty of impression chances which could have been made

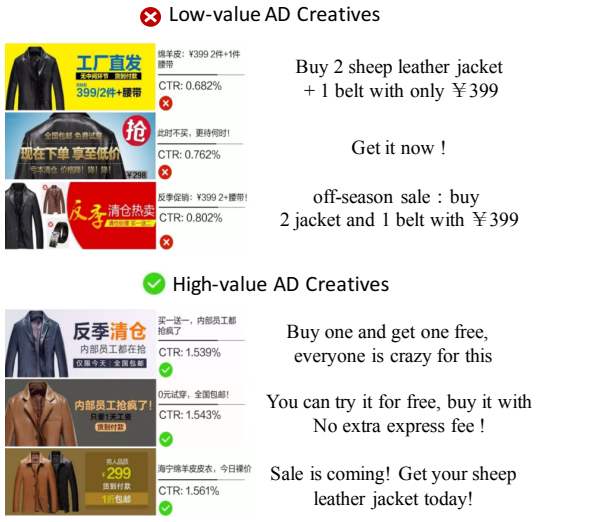

Figure 1: Ad creatives cases (adopted from Tencent Creative College[1]). Each creative includes an image and a slogan. Their averaged click-through-rates (CTRs) are also presented. Creatives with better design of color, layout , and style can attract more user clicks and buying activities. Based on the observation, this paper studies how to evaluate the performance of ads by their content before showing to users online. The top evaluated ads are then displayed online to get most revenue.

[1]https://idea.qq.com/college

CIKM '19, November 3–7, 2019, Beijing, China

ACM ISBN 978-1-4503-6976-3/19/11…$15.00
https://doi.org/10.1145/3357384.3357813

better use of. In fact, there is a large proportion of new ads that could not accumulate enough clicks or conversions for convergence and hence fail in cold start process, which means their historical impressions are completely wasted. It becomes even worse with the number of ad creatives in an ad plan increasing. For example, an advertiser plans to promote its product on the advertising platform. It prepares M images of the product and N slogans for the promotion ($M, N \leq 10$ for most ads), and totally $M \times N$ new ads are created in this ad plan. Without any user activities and prior knowledge of the ad performance, the new ads will be explored one by one in random order until one of them becomes "matured" and meets the expectation of the advertiser. As a result, the larger $M \times N$ is, the more impression chances it will cost to explore the superior ads with the highest value in the ad plan.

In the paper, we focus on the following problem: *given an ad plan containing hundreds of new ads, how to promote the ads with the highest potential profit among numerous creatives, with minimal trial-and-error cost during the cold-start phase.*

We observe that the ad creative content determines whether the user will be attracted or not (as shown in Figure 1). Based on the observation, some previous works [12, 18] propose to extract the content features from ad images and then incorporate the content features into the CTR prediction model. In this way, the content features are expected to improve the CTR prediction accuracy when the ID features are "unmatured" in cold start process. However, this pipeline still needs to explore the new ads online one by one, and thus it could hardly handle the case with hundreds of new ads. Moreover, limited by the online system capacity, they can not employ the complete large scale image/slogan feature extractors in an end-to-end training framework.

Instead of the methods above, we propose a "pre-ranking stage", which is executed offline, before ad targeting and online ranking. When one advertiser submits all his creatives, the prepared offline pre-ranking model determines inferior and superior creatives. The superior creatives will be fed into the online ranker under each request to make final decisions. The pre-ranking stage have the following advantages: 1) Since the stage can be executed offline, we can use complete large-scale models (deep neural networks) to extract content features. We can also provide enough time to generate well-trained CNN networks. 2) The powerful deep features, compared with light-weighted features such as words tags used in the online ranker, provide more detailed and discriminative information, and improve the platform revenue significantly. 3) By accumulating historical data, we model the training as learning-to-rank problem, and avoid predicting CTR straightforward. The model requires no user data. 4) The most importantly, deep neural networks have well generalization and perform very well on transferring learned knowledge on unseen samples. The offline evaluation model provides "matured" features on new ads, and thus extremely mitigates the online cold start problem.

The main contributions of the paper are summarized as the following:

- First, we propose the ad pre-ranking stage, which can be flexibly plugged into the advertising system. The stage predicts ad rankings by powerful deep neural networks. The top-ranked candidates, which are considered as superior ad creatives, will be fed into the online system and explored in priority. By the offline pruning strategy, the wastage of impression chances could be reduced significantly.
- Next, we propose Pre Evaluation of Ad Creative Model (PEAC), the specific deep neural network (DNN) model used in the pre-tanking stage. It utilizes images, slogans, OCRs and context features to evaluate the content. Meanwhile, we propose a pairwise method for training it. The DNN model is trained to learn the relative ordering instead of the CTR. It reduces the difficulty in model training and fits our task well.
- Finally, we verify the proposed PEAC through both offline and online experiments. In offline experiments, the ranking results of the pairwise model outperform all of the baseline methods. In online AB test, the increasing effective Cost-Per-Mile (eCPM) and success rate in cold start also highlight the success of the proposed system. The proposed method has been deployed in the large-scale online advertising system at ByteDance.

2 RELATED WORK

Our work is related to CTR prediction in recommendation/advertising system. In this section, we will briefly review some representative and related works.

CTR Prediction. Click prediction for online ads is the core task of online advertising system, which has a direct effect on the advertising revenue. Consequently it attracts extensive attention in both research and industry community. Generally, the problem is formulated as a click prediction task, in which both the features of ad and user in one impression are fed into the model to predict whether the user will click the ad or not. While in traditional works logistical regression [7, 17] and decision trees [14] have been widely applied, the most recent works [9, 24] try to employ the popular deep neural networks to improve the prediction accuracy. Despite various models, most of the existed CTR prediction methods heavily rely on the high-dimensional sparse ID features (both ad IDs and user IDs). As discussed above, ID features need a considerable number of impression chances to be "matured" hence the methods suffer from the cold start problem.

To address the cold start problem, early works [1, 4, 10] utilizes contextual information such as hierarchical ad category and historically related ads to help CTR prediction for new ads. Thanks to the recent development on convolutional neural networks (CNN) and computer vision, the ad images which contains rich information and largely determines user behavior, have been tried to be incorporated into the CTR predictors. [18] is the first to use CNN model as image feature extractors, and shows such learnable image representation achieve better performance in CTR prediction than handcrafted features [2, 8]. Furthermore, [6, 11] involve both image features and user features in an end-to-end network training to predict user click or not.

We differentiate the paper from the aforementioned works in these aspects: 1) our task is different from the online CTR prediction task, whose target is to evaluate the new ads offline without requirement of impression chances or user features; 2) compared to the previous works predicting the absolute CTR or click, the paper proposes a pairwise loss to train a CNN model to predict the

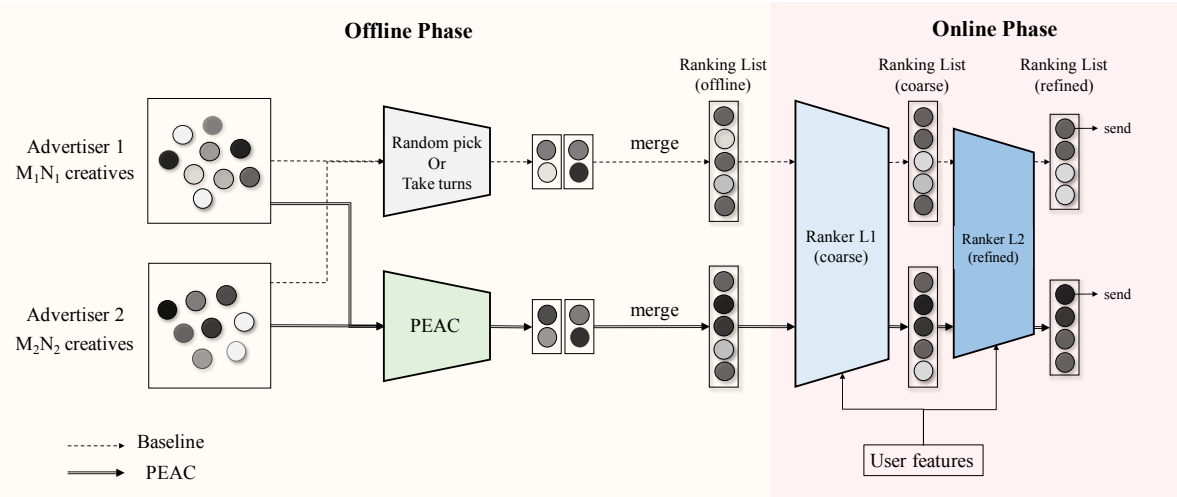

Figure 2: The comparison between the exiting method and the proposed PEAC in online advertising system. Darker circles represent better creatives with higher potential profits.

relative order of the ads in one ad plan. 3) our model is implemented offline, which is eventually a supplementary stage for online CTR prediction.

3 OVERVIEW OF PEAC

In the section, we take an overview of the proposed PEAC system. We will present a standard online advertising system and the augmented system with the introduction of PEAC. We will focus on our proposed offline pre-evaluation model.

Figure 2 shows a typical advertising system. We are showing two versions of the system in the same figure: the baseline version (the computation follows the dotted line) and the improved version with PEAC component (the computation follows the solid line). In this system, advertisers provide ad materials with M product images and N ad slogans in one ad plan. They form $M \times N$ creative candidates, denoted by circles in Figure. 2, where deeper color means better online performance. In the existing baseline pipeline, the newly created ads are fed into the two-stage online predictors (a coarse ranker L1 and a refined ranker L2) in a random or in-turn order, in which their ID features will be learned through thousands of impression chances. For each advertiser, we pick at most k creatives, and merge all of them to the online rankers. In each future impression, only the top-ranked ad will be sent to users[1].

There are some obvious drawbacks in the existing pipeline:

- **Non-optimal Performance**. Without the understanding of the content of ad creatives, all creatives with various potential profits are treated equally. If the best one falls behind in the random order, it results in the non-optimal advertising performance for both advertisers and platforms.
- **High Cost**. The inferior ads are not pruned off early and they will occupy plenty of impression chances in the cold start process. It leads to an increasing trial-and-error cost in time and advertising income.
- **Lack of Generalization**. With only ID features, the advertising system suffers from less generalization ability. Even when the advertiser uploads an exactly same creative which has been displayed before, it will be assigned a new ID and learned from the beginning.

As mentioned before, in this paper we propose a "pre-ranking stage" to improve the cold start problem (the lower branch in Fig.2). When one advertiser submits all his creatives, the prepared offline pre-ranking model determines inferior and superior creatives. Creatives with high potential profits are sent to online rankers in priority. Since the progress is executed offline, we can train a powerful deep-neural-network-based model, and formulate the problem as a learning-to-rank problem. Compared with the former pipeline, it navigates the problems in the following aspects:

- **Better Performance**. The model learns the relative ordering of ad creatives by their potential values. So the best creatives are expected to be impressed online in priority.
- **Less Cost**. The inferior ads are pruned off at the very beginning and thus no impression chance is wasted on these ads. On the contrary, more impression chances could be offered to the ads with high potential value, making them "matured" faster. It will significantly save the time and income cost in cold start process.
- **Meaningful Generalization**. The model is able to learn some generalized knowledge of determining good creatives. The more historical creatives it is trained by, the more generalization ability it has. It will be a good complement for the ID feature-based online rankers.

[1] in practice, online rankers take candidates from two individual ranking lists, one for matured creatives and the other for newborn ones. In this paper, we only discuss the list for the newborn creatives.

- **Guidance to Creative Design**. Knowledge learned by the model may guide the advertisers to design good ad creatives. Even a generative system could be built to edit the ad creatives automatically. This part is beyond the main contributions of the paper and we leave it to future works.

4 PEAC MODEL TRAINING

In this section we introduce the proposed PEAC offline evaluation model. There are several aspects to be discussed: (1) how to define a "superior" creative, (2) in which way we train the ranking model, and (3) the details of the model.

4.1 Training Framework

To train an offline evaluation model, a straightforward way is to predict Click-Through Rate (CTR) and Conversion Rate (CVR). However, there are several problems: (1) CTR is not intuitive, for example, it is not easy to figure out whether a creative of 3.5% CTR is definitely superior to another with 3.4%. (2) Without user features, it is hard to train a model to predicts CTRs. Besides, CTRs do not outline quality of creative content. (3) CTR is not well matched with the behaviors of the online system. Some creatives show misleading titles or images to obtain high CTRs, however, they cannot improve, and sometimes even decrease the revenue. Our learning target should consider online system behaviors.

We tackle the first two problems by loosing the training constraints. We propose a pair-wise training framework [3]. As shown in Fig.3, two creatives are paired $(\mathbf{x}_i, \mathbf{x}_j)$. We extract their images $(\mathbf{I}_i, \mathbf{I}_j)$, slogans $(\mathbf{T}_i, \mathbf{T}_j)$ and OCRs $(\mathbf{O}_i, \mathbf{O}_j)$, feed forward to our offline evaluation model, and calculate the corresponding ranking scores s_i, s_j. With given targets y_i, y_j, we enlarge the difference of the two scores by a Binary Cross Entropy loss:

$$loss = -(y \log \sigma(s_i - s_j) + (1 - y) \log(1 - \sigma(s_i - s_j))) \tag{1}$$

$$y = \begin{cases} 1, & \text{if } y_i > y_j \\ 0, & \text{if } y_i < y_j \end{cases} \tag{2}$$

where σ is the Sigmoid function. We found it fits our task well, and provides better performance.

For the third problem, we propose a target which better describes online system behaviors and strengthens differences of creatives. For each pair that has been chosen by the online ranker L1, we summarize the amount that only $\mathbf{x}_i$ or $\mathbf{x}_j$ is chosen by the online ranker L2, denoted as n_i^{ij}, n_j^{ij}. The superscript indicates the two counters only works for the pair (i, j). For each request, we first check if both $\mathbf{x}_i, \mathbf{x}_j$ appear in the coarse ranking list. If so, we check if only one of them (e.g. $\mathbf{x}_i$) appears in the refined ranking list. If it appears then we count once for n_i^{ij}.

By aggregating n_i^{ij} and n_j^{ij} of all the requests, we obtain y_i and y_j. The advantages of such training target is two-fold, on the one hand, the difference of y_i and y_j is always significant, a typical pair is about 1000 vs 10, it is easy to define whether $\mathbf{x}_i$ is significantly superior to $\mathbf{x}_j$. On the other hand, the target somehow learns online system behaviors, better $\mathbf{x}_i$ promoted by our model is also easier to be promoted by the online system (we also compare other ways of generating y_i, see Sec.5.1 for details.).

Since advertising system is extremely complex and its behavior is somehow undetermined, the results of online rankers are inevitably noisy. So we need to filter some noisy data. In this paper, we select pairs by the following principles:

- **Saliency**. We only consider pairs whose $|y_i - y_j| > h$ and $y_i/(y_i + y_j) \geq h_u$ or $y_i/(y_i + y_j) \leq h_l$. Other pairs are considered as not salient samples.
- **Unbiasedness**. Impression counts of both creatives should exceed h_s, such constraint requires them to be verified by most users and makes results stable.
- **Consistency**. Creatives with CTR $> h_c$ will be discarded. Those samples are considered as clickbait. We only use normal creatives to learn our model.

In the online advertising system, advertisers propose multiple ad sets and each of them contains several creatives. One ad set denotes one kind of inventory, like "Adidas" or "Nike". In this paper we only define pairs on creatives belonging to the same ad set, and pairs among various ad sets are considered as uncomparable.

Using online ranker predictions somehow increase the risk of getting biased result, and we employ pruning strategy to remit the influence: some weak creatives according to their performance will be pruned, replaced by unexplored ones.

4.2 Model Design

The design of the model is also critical. Even the original creatives have provided images and titles, there are also other kinds of important descriptions embedded in images. Following the principle that we should utilize abundant features, we use an OCR extractor, which follows a SSD framework [22], to capture such features (as shown in Fig.3). For images, we use the MobileNetV2 [19]. Slogans and OCR representations are fed into two individual Character CNNs [23]. Then, features of different modalities are concatenated and transformed by the proposed tree-like dynamic fully-connected layer (TDFC, see below), and produce the output score s.

4.3 Tree-like Dynamic Fully Connected Layers

In online advertising system, the industries of creatives are critical. For examples, for a creative about games, users always pay attention to the images. While for creatives of medicines, users are easily affected by the slogans. This implies that for creatives belonging to various industries, multi-modal features should be combined adaptively.

In this paper, we propose the Tree-like Dynamic Fully-Connected layers (TDFC). First, we use one FC for each industry, dynamically selecting the corresponding layer for training (Thanks to the dynamic graph based learning framework[2]). In both training and test phases, for a coming pair belonging to industry $t \in \{1, 2, 3 \dots T\}$ (in one ad set, creatives belong to the same industry), we use the output of the t-th fully-connected layer to obtain its score. In this way, features can be treated differently for various industries. This is called the Dynamic Fully-Connected layer (DFC).

However, the naive DFC has two drawbacks: 1) each of the FCs is only optimized by $1/T$ samples and 2) FCs share no knowledge

[2] www.pytorch.org

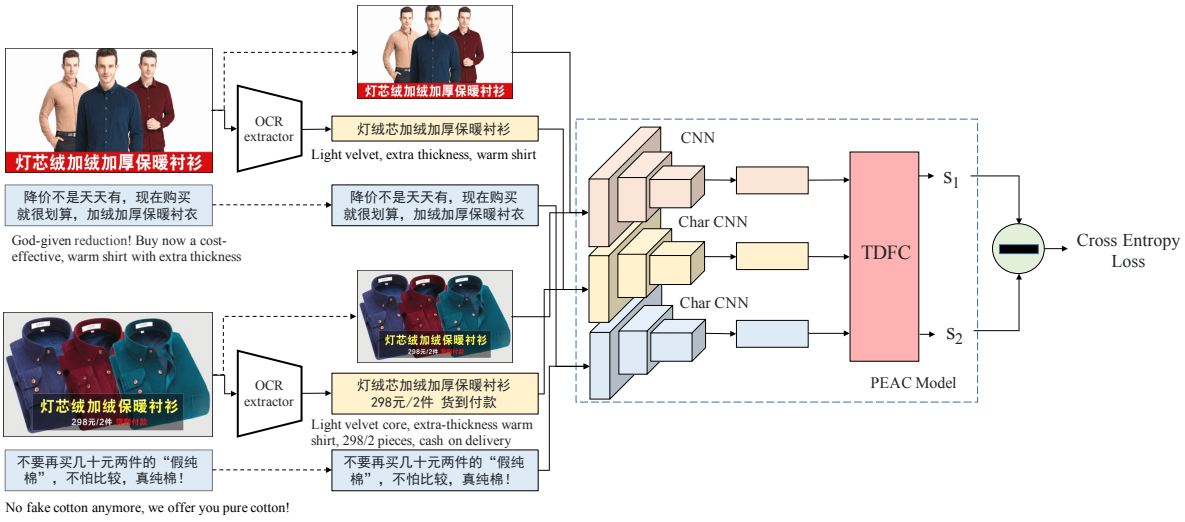

Figure 3: The proposed PEAC with the pair-wise training method. For two samples belonging to the same pair, their images, titles and OCR representations are fed into the CNNs (for title and OCR we use character CNN [23]). Then concatenated features are passed to a fully-connected layer and scores for the ad creatives are produced. Slogans are translated for non-Chinese speakers.

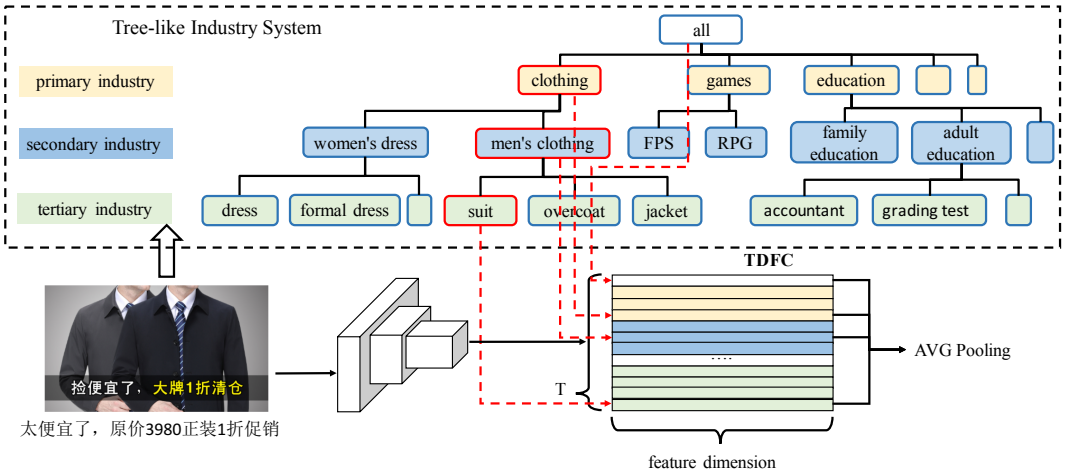

Figure 4: The proposed TDFC. For each sample we mean the scores generated by its primary/secondary/tertiary-industry FCs, in training as well as inference phases.

with each other. Considering that the industries are forming a tree-like structure, we could aggregate the tree-like outputs. As shown in Fig.4, the creative actually has three industry tags for primary industry, secondary industry and tertiary industry. For each sample, its score is calculated as the mean of the outputs corresponding to its three-level industries. Now the sharing knowledge can be captured by primary/secondary FCs and the primary FCs are ensured to be optimized by abundant samples. This is the definition of TDFC.

Intuitively, we need T FCs to construct a TDFC, in practice, however, TDFC is implemented by a single large FC with the shape of (d, T) where d refers to feature dimension. In the inference phase no dynamic graph or frameworks are needed, we forward the TDFC for once and it yields T scores simultaneously, then we mean the scores at specific locations. The TDFC costs nearly no extra time than a single FC. Note there is a leading FC for all industries (white in Fig.4), so for each sample its score is average pooled by $1 + 3 = 4$ scores.

Table 1: Offline evaluations on point-wise & pair-wise training.

Training ways	NDCG	Top10
CTR Prediction	66.1	62.9
CTR Pairs	67.0	63.3
L1 n_i/n_j	67.2	63.4
L2 n_i/n_j	67.8	63.7
L2 to L1 n_i/n_j	**68.2**	**64.1**

4.4 Implementation Details

We choose MobileNetV2 [19] as our CNN model. For character CNNs, we use kernels of size 1-8, and concatenate the corresponding features. The number of channels is 256. During training, the batch size is set to 1280 (mainly implemented by gradient accumulation), image inputs are randomly cropped to 224×224, before that, scale jittering [16] is used.

5 EXPERIMENTS

In this section, we evaluate our proposed PEAC in the online advertising system. The experiments are mainly composed of offline evaluations and online evaluations. We evaluate various models and ways of making pairs on offline test dataset, and show A/B test results online.

5.1 Offline Evaluations

Datasets. As mentioned above, in this paper we only define pairs on creatives belonging to the same ad set, and pairs among various ad sets are considered as uncomparable. We collect 4 millions pairs of 2 months for training data. Original image sizes vary from 300×900 to 480×640, but all images are pre-resized to 256×256 for running speed. One epoch takes about 1 day on a single 1080Ti, and we train $2-3$ epochs before convergence.

In the test phase, we collect creative lists of each ad set for half a month, and predict the scores for creatives. The ground truth rankings are determined by their real-world impression counts. Since the bid among one ad set keeps the same, the impression count is directly related to platform revenue in our oCPM system. Besides, online rankers always promote superior creatives to be impressed, so higher impression counts present better creatives. However, we do not use impression counts in the training phase, since posterior data has a small amount.

Metrics. For each list, we calculate the overlap of the top 10 predictions between models and ground truth. We also calculate Normalized Discounted Cumulative Gain (NDCG [21]) for better understand the difference of ranking results. The top 10 overlap is defined as intersection-over-union of predictions of models and the ground truth:

$$\frac{S_p \cap S_g}{S_p \cup S_g} \tag{3}$$

where S_p denotes the set of predicted top 10 items, and S_g is the set of ground truth top 10 items. After calculating NDCG/top10 for each list, we average the results on all lists.

Point-wise vs. Pair-wise Training. In Table.1, we show results of different training methods (for better view, we normalize NDCG/Top10 to 0-100%). "CTR Prediction" denotes the point-wise method which directly predicts CTRs (using historical data, as classification task).

Table 2: Performance of single branches & multi-modal features.

feature branch	NDCG	Top10
title	60.2	61.0
img	60.3	61.0
title+img	66.7	63.7
title+img+OCR	68.2	64.1

Table 3: Comparison on model design techniques.

CNN Network	Slogan/OCR network	CNN fine-tuning	TDFC	NDCG	Top10
MobileNetV2	LSTM			62.9	59.3
MobileNetV2	LSTM	✓		66.9	61.5
MobileNetV2	CharCNN	✓		68.2	64.1
MobileNetV2	CharCNN	✓		68.9	64.2
ResNet-50	CharCNN	✓		69.4	64.2
ResNet-50	CharCNN	✓	✓	**69.9**	**64.6**

Learning point-wise metrics is essentially more difficult than learning pair-wise rankings. Thus, it obtains inferior results. For the pair-wise framework, we also try various methods on making pairs. "CTR Pairs" makes pairs by predicted CTRs of online rankers, however, this method is easily affected by clickbait samples. "L1/L2 rank" means when both creatives of a pair appear in the ranking list of the corresponding online ranker, if $\mathbf{x}_i$ leads $\mathbf{x}_j$, we count once for n_i^{ij}, vice versa. Then, "L1 to L2 rank" denotes the method used in this paper as described in Sec.4. The model architecture is fixed: we use MobileNetV2 [19] for image and Character CNN [23] for slogan/OCR.

The first observation is that pair-wise training procedure makes learning easy. By loosing point-wise method to CTR based pairs, the results are improved, which verifies the claim in Sec.4 that the proposed training target improves the performance. The using of n_i/n_j improves the performance to 67.8/63.7, which implies it better describes online system behaviors, and introduces less noisy data. Among two levels of online rankers, using predictions of any of them achieves competitive results, however, using the difference between two models achieves the best results. "L1 to L2" ranking is the best choice to make pairs. The two rankers are actually trained for the same target (CTR prediction) but in different scale. Ranker L1 aims to recall superior creatives, and L2 aims to precisely promote the best one. "L1 to L2" ranking actually highlights difference between the two models, but "L1/L2" ranking only captures fine distinction within one model.

From the above discussion, we conclude the following points:

- Compared with point-wise method, learning pair-wise ranking relationship is a better choice.
- Single CTR poorly determines which creative is better. Instead, the proposed y_i/y_j better describes online system behaviors.
- Single online ranker may provide noisy ranking. However, the difference of the two rankers highlights difference of creatives.

Multi-modal Features Now we evaluate the performance of each branch of the multi-modal, as well as the aggregated features in Table.2. We found single image or title features provide significant inferior performance, and only the combination of them makes sense. The most important insight is the decouple of embedded cues:

by employing OCR features, we obtain significant improvement, +1.5% and +0.4% on NDCG and Top10. We can learn that:

- We should employ multi-modal features, which provide comprehensive cues.

Details of Model Design. In Table.3 we show results of various model architectures. First, the network requires more knowledge than content (the definition of content and design can be found in Sec.5.4), which can be inferred by the comparison of w. and w/o. fine-tuning. Such observation is discussed in the following section. The following improvement comes from choice of image and slogan/OCR models, Character CNN outperforms LSTM [20] on both effectiveness and training efficiency, and ResNet-50 [13] learns more accurate mapping function than MobileNetV2. Since the underlying knowledge of creative preference is not intuitionistic, deeper CNN learns better mappings and performs better. The performance of LSTM and CharCNN is an interesting observation, which probably reminds that wortstellung is always omitted in the short impressions but the occurrence of some keywords matters.

In the experiments, tree-like dynamic fully-connected (TDFC) layers provide adaptive decisions for various industries. Compared with the results of a single FC, TDFC improves the results by 0.7% and 0.6% on NDCG and Top10, relatively. Note even though the dynamic-graph-based framework is need for training, in the deploy phase, TDFC requires no dynamic graph. In the pre-ranking stage the whole of T outputs are generated, but we just pool several specific outputs outside the computation graph.

We learn the following conclusions:

- The model prefers deeper networks for image features, but more plain networks for title/ocr. This suggests that in display advertising, image cues and occurrence of keywords are more important.
- In various categories, cues are embedded in different ways, with TDFC, we can better describe these cues, and adaptively decode them for better performance.

5.2 Online Evaluations

Now we use online A/B test to further verify the proposed method.
Data. For online experiments, we take real-world A/B test. For each kind of method, we use it to predict top10 best creatives in an ad set. These creatives are sent to the online rankers and further be sent to users. We summarize metrics for each creative and show the average. Each creative will be observed for 10 days and later new conversion will not be considered (in general, a creative can keep active for a week). Each A/B test is executed for a month.
Metrics. We use the following metrics for online experiments: CTR, CVR, eCPM and Convert 10/20/30 Rate. CTR, CVR and eCPM are calculated as general Click-Through Rate, Conversion Rate and effective Click-per-Mille. Convert 10/20/30 Rate counts how many creatives achieve more than 10/20/30 conversions after being chosen by our model, which eventually measure the success rate in cold start.

The results are shown in Table.4, where each block refers to one A/B test and we show the relative increment. "Random Pick" means we use no offline models[3].

[3]In online systems, we could employ some Exploration-Exploitation-based methods, such as Thompson sampling [5], to yield better results. However, these methods require immediate feedbacks to make adjustments. In this paper, such methods have been used in online rankers but cannot be used in our offline model. The most basic baseline is the method of random pick. We treat the PEAC (L2 rank) as a better baseline.

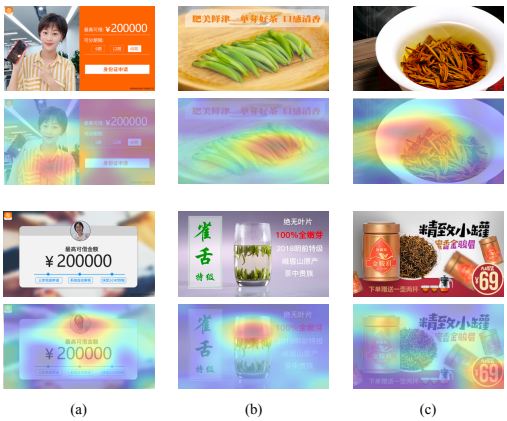

Figure 5: Visualization of response in our model. Each column presents one pair of creatives. All of the upper creatives obtain better performance than the lower ones.

The first A/B test demonstrates that the proposed PEAC learns strong cues only by the creatives themselves and it performs significant better than random pick method on most metrics except for Convert 10[4]. We also compare different ways of making pairs, "L1 to L2 n_i/n_j" obtains the best results. Such results accord with the observations in our offline experiments.

The proposed TDFC also improves the performance significantly. However, the gain of TDFC is less than "L1 to L2 n_i/n_j". This suggests that the training target design is more critical than model design.

From the above discussion, we conclude the following points:

- PEAC improves cold start method significantly. According to the online metrics, it essentially improves online platform revenue ($CTR \times CVR$), which is critical for online auction advertising system.
- From the Convert 10/20/30, creatives picked by PEAC are more probable to be "matured" and occupy less spots, which saves time and income cost in cold start process.
- PEAC learns well generalization, since in our offline/online experiments, test creatives have no overlap with training samples. It keeps knowledge for new creatives.

5.3 What Knowledge Is Learned by The Model?

Following the above observation, we wonder what knowledge has been learned by our model? To this end, we visualize the response of our image CNN by a simple technique: we mask each region of activations on feature maps, and calculate the drops of result scores. The resulting visualization is shown in Fig.5.

With the visualization results, we have some interesting observations. From the first sample, our model prefers humans in creatives. Even the loan amount is obvious in the lower one, our model still

[4]This is not a drop, since in fact many creatives are calculated in "Convert 20/30" and makes there are less ones in "Convert 10".

Table 4: Online evaluations.

Method	CTR(Δ)	CVR(Δ)	Convert10(Δ)	Convert20(Δ)	Convert30(Δ)	eCPM
no PEAC (Random Pick)	-	-	-	-	-	-
PEAC (L2 n_i/n_j)	+12.2%	+8.34%	-1.11%	+1.03%	+4.25%	+6.05%
PEAC (L2 n_i/n_j)	-	-	-	-	-	-
PEAC (L1 to L2 n_i/n_j)	+12.52%	+9.01%	+1.32	+3.88%	+5.69%	+6.59%
PEAC (w/o. TDFC)	-	-	-	-	-	-
PEAC (w. TDFC)	+5.77%	+4.31%	+0.62%	+0.77%	+2.10%	+2.90%

这裤子老公穿了三天都不舍得脱，结果又买了两条！(My husband has wearing it for three days, and I have to buy two more!)
别犹豫！这男士休闲裤质量很好，卖完为止！(Check this qualified casual pants, it is selling out!)
专业针对高三学生，解答高考提分方案!一定要看看！(Professional for high school students, improves CEE scores. You must check this!)
家里有高三考生，家长该怎么让孩子少走弯路？(Teach your pre-college child pass the college entrance examination (CEE).)
家长须知！高考备考，消灭难点攻破考点,提分攻略！(Prepare the exam, pass the test, improve the score!)
本区实时天气预报，想几点看就几点看 (Real-time local weather forecast, check any time you want)
今日本地强降雨天气预报！出行注意安全 (Local heavy rainfall today! Be careful outside)
本地预测未来15天天气，精准预报未来几点几分下雨(Local forecast for the next 15 days, accurate to minutes)
在本地装修前一定要看这家的报价，不后悔！(I bet you will regret if you miss this offer!)
在本地装修新房必看，这家装饰公司真不错！(You are not willing to miss this decoration company!)
太赚了！装修预算合理花费，透明报价，拒绝装修小猫腻！(Reasonable cost of renovation budget, transparent quote, no tricks.)

Figure 6: The prediction results of our model. Each column shows the ground truth ranking of a list of creatives (from top to bottom we show best and worst ones). For each creative, we mark its predicted ranking at the right. From the results, our model learns knowledge on how to select better creatives. Slogans are translated for better reading.

focuses on the human head. In most samples, more attentions on items always provide better results. One unexpected observation is that the model "hate" price tags. In the last group, the price of the lower creative is highlighted, however, it leads to worse results. The reason may be that such a price tag can not tell whether or how it is economically related to the item. Instead, it diffuses users on item itself.

Table 5: Comparison of content and design embeddings.

Method	NDCG	Top10
Design embeddings	68.3	63.7
Content embeddings	67.5	63.5
Combination	69.9	64.6

We also visualize the predicted lists in Fig.6. The model has the knowledge to distinguish better creatives.

5.4 Disentangle of Content and Design.

In this section we explore more underlying knowledge. For example, given an image, two kinds of information are embedded: the content embeddings and the design embeddings. The content embeddings present what the image show, what kind of inventories? Are there kids in a toy creative? The design embeddings, however, describe how the image is shown. What kind of color scheme is used? Where the description is located? We can design experiments to show which of them is more critical for creatives.

To this end, we need to disentangle the two kinds of embeddings separately. First, we try to suppress the content. We utilize a very simple technique to achieve this goal: use a 50×50 gaussian kernel and blur the image for several times. In such an image humans can hardly figure what is shown, while the overall layout and color are preserved.

The method of suppressing design embeddings is to freeze weights of the image branch. Since the pre-trained model is only trained to learn content of ImageNet [15], it can provide limited knowledge on design. So we just use it as feature extractor and train the rest part.

In Table.5 we show the comparison of the two kinds of embeddings. The first observation is that creatives are determined by both of the embeddings, using only one of them obtains inferior results. Then, design embeddings, as low-level cues, play more important roles in offline evaluations. Even this is somehow unexpected, it can be explained: most users prefer using less time on choosing to click the creatives or not, so cues that can be captured in seconds make sense. This conclusion is also valuable for advertisers or designers: **better design is more important than better content**.

6 CONCLUSION

In this paper, we study the cold start problem of ad evaluation and selection. We use a pre-ranking stage to employ deep neural networks, and propose PEAC, a novel method to evaluate ads before they were displayed and attracting any clicks. We propose to train the model pair-wisely and a procedure to construct positive/negative creative pairs. We have implemented and deployed the system in ByteDance Advertising Products. In the online testing with real ads and user traffic, the proposed PEAC improves critical metrics significantly, and therefore leads to large revenue gain. Through our experiments, we explore the key factors in creating good ads, which provide insights for advertisers and designers.

REFERENCES

[1] Deepak Agarwal, Rahul Agrawal, Rajiv Khanna, and Nagaraj Kota. 2010. Estimating rates of rare events with multiple hierarchies through scalable log-linear models. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 213–222.

[2] Javad Azimi, Ruofei Zhang, Yang Zhou, Vidhya Navalpakkam, Jianchang Mao, and Xiaoli Fern. 2012. The impact of visual appearance on user response in online display advertising. In *Proceedings of the 21st International Conference on World Wide Web*. ACM, 457–458.

[3] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. 2015. Learning to Rank using Gradient Descent. In *ICML*.

[4] Deepayan Chakrabarti, Deepak Agarwal, and Vanja Josifovski. 2008. Contextual advertising by combining relevance with click feedback. In *Proceedings of the 17th international conference on World Wide Web*. ACM, 417–426.

[5] Olivier Chapelle and Lihong Li. 2011. An Empirical Evaluation of Thompson Sampling. In *NIPS*.

[6] Junxuan Chen, Baigui Sun, Hao Li, Hongtao Lu, and Xian-Sheng Hua. 2016. Deep ctr prediction in display advertising. In *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 811–820.

[7] Haibin Cheng and Erick Cantú-Paz. 2010. Personalized click prediction in sponsored search. In *Proceedings of the third ACM international conference on Web search and data mining*. ACM, 351–360.

[8] Haibin Cheng, Roelof van Zwol, Javad Azimi, Eren Manavoglu, Ruofei Zhang, Yang Zhou, and Vidhya Navalpakkam. 2012. Multimedia features for click prediction of new ads in display advertising. In *KDD*. ACM, 777–785.

[9] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*. ACM, 7–10.

[10] Kushal S Dave and Vasudeva Varma. 2010. Learning the click-through rate for rare/new ads from similar ads. In *SIGIR*. ACM, 897–898.

[11] Tiezheng Ge, Liqin Zhao, Guorui Zhou, Keyu Chen, Shuying Liu, Huiming Yi, Zelin Hu, Bochao Liu, Peng Sun, Haoyu Liu, et al. 2017. Image Matters: Jointly Train Advertising CTR Model with Image Representation of Ad and User Behavior. (2017).

[12] Tiezheng Ge, Liqin Zhao, Guorui Zhou, Keyu Chen, Shuying Liu, Huimin Yi, Zelin Hu, Bochao Liu, Peng Sun, Haoyu Liu, Pengtao Yi, Sui Huang, Zhiqiang Zhang, Xiaoqiang Zhu, Yu Zhang, and Kun Gai. 2017. Image Matters: Visually modeling user behaviors using Advanced Model Server.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. *arXiv preprint arXiv:1512.03385* (2015).

[14] Xinran He, Junfeng Pan, Ou Jin, Tianbing Xu, Bo Liu, Tao Xu, Yanxin Shi, Antoine Atallah, Ralf Herbrich, Stuart Bowers, et al. 2014. Practical lessons from predicting clicks on ads at facebook. In *Proceedings of the Eighth International Workshop on Data Mining for Online Advertising*. ACM, 1–9.

[15] Deng J., Dong W., Socher R., Li L.-J., Li K., and Fei-Fei L. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

[16] Simonyan Karen and Zisserman Andrew. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. abs/1409.1556 (2014).

[17] H Brendan McMahan, Gary Holt, David Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, et al. 2013. Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1222–1230.

[18] Kaixiang Mo, Bo Liu, Lei Xiao, Yong Li, and Jie Jiang. 2015. Image Feature Learning for Cold Start Problem in Display Advertising. In *IJCAI*.

[19] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[20] Hochreiter Sepp and Schmidhuber Jürgen. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[21] Hamed Valizadegan, Jin Rong, Ruofei Zhang, and Jianchang Mao. 2009. Learning to Rank by Optimizing NDCG Measure. In *Advances in Neural Information Processing Systems 22*, Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, and A. Culotta (Eds.). Curran Associates, Inc., 1883–1891.

[22] Liu Wei, Anguelov Dragomir, Erhan Dumitru, Szegedy Christian, Reed Scott, Fu Cheng-Yang, and Berg Alexander C. 2015. SSD: Single Shot MultiBox Detector. (2015).

[23] Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level Convolutional Networks for Text Classification. In *NIPS*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett (Eds.). Curran Associates, Inc., 649–657.

[24] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 1059–1068.