Image Feature Learning for Cold Start Problem in Display Advertising

Kaixiang Mo[†], Bo Liu[†], Lei Xiao[‡], Yong Li[‡], Jie Jiang[‡]
[†]Hong Kong University of Science and Technology, Hong Kong, China
[‡]Tencent Inc., Shenzhen, China
[†]{kxmo, bliuab}@cse.ust.hk, [‡]{shawnxiao, nickyyli, zeus}@tencent.com

Abstract

In online display advertising, state-of-the-art Click Through Rate(CTR) prediction algorithms rely heavily on historical information, and they work poorly on growing number of new ads without any historical information. This is known as the the cold start problem. For image ads, current state-of-the-art systems use handcrafted image features such as multimedia features and SIFT features to capture the attractiveness of ads. However, these handcrafted features are task dependent, inflexible and heuristic. In order to tackle the cold start problem in image display ads, we propose a new feature learning architecture to learn the most discriminative image features directly from raw pixels and user feedback in the target task. The proposed method is flexible and does not depend on human heuristic. Extensive experiments on a real world dataset with 47 billion records show that our feature learning method outperforms existing handcrafted features significantly, and it can extract discriminative and meaningful features.

1 Introduction

Online advertising is a rapid growing multi-billion dollar business. Many IT companies like Google, Tencent and Baidu have large percentage of revenue coming from online ads. As more and more people prefer shopping online, many internet sellers are increasingly hoping to show their ads to online users. Image ads enjoy big advantages because they are compact, intuitive and easy to understand. In this paper, we focus on image ads in display advertising.

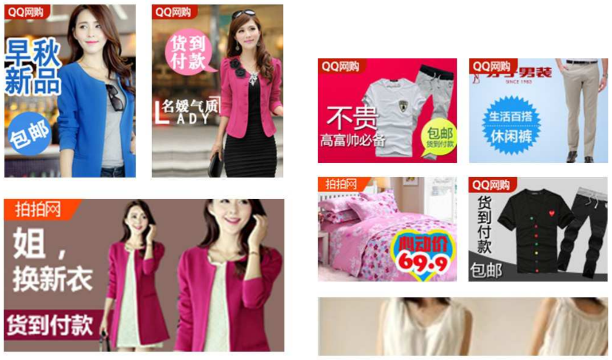
Accurately predicting the Click Through Rate(CTR) of selected ads is the core task of ad-networks. Sellers pay the ad-network when a user clicks on the ad, so showing the most attractive ads to target customers not only increase the revenue but also improve the user experience. State-of-the-art click prediction algorithms rely heavily on historical information, for example historical CTR, id of the ads and categories of the ads. Since the CTR of mature and stable ads do not change much, these algorithms work quite well on old ads. However, they are not suitable for new ads without sufficient historical information.

New ads are extremely important. In fast changing markets, users are easily tired of old ads and sellers need to frequently update their ads. As a result, most ads have short life expectancy. Besides, more and more new sellers are hoping to show their ads via ad-networks. In this situation, new ads constitute a large proportion of all the ads. If the click prediction system does not pay enough attention to new ads, the prediction system can not accumulate new user feedback on new ads, eventually the system will go into self-destructive circle.

In order to deal with the cold start problem in new image ads, some existing prediction systems used image features to identify ads with similar characteristics and thus to predict the CTR of new ad images. Due to privacy reason, some advertising systems are not allowed to use personal information. In this paper, we focus on learning better image features for advertisements, and we leave the personalized user taste of ad images for future discussion.

Existing image features used in image display ads are mainly handcrafted features. They are designed from various perspectives such as lighting, color, sharpness, blur, subject quality, rule of third, simplicity, visual weight, dynamics, color emotions. Some other handcrafted image features are designed especially for object recognition tasks, such as SIFT features. However, these handcrafted features do not work well on image ads. Firstly, they are not suitable for the click prediction task. These visual features are not especially designed for the click prediction task. They are mainly low level features with limited representation capability and very few of them can accurately capture key factors for the click prediction task. Secondly, they are inflexible. The key factors leading to click action may be different from task to task and from time to time. Taking miniskirt as an example, the key factor for attractiveness may be simply trending color in 1990's, but later becomes complex pattern or even fashion design which can no longer be captured by a color feature. What's worse, new handcrafted features rely heavily on human heuristic. As a result, they are very hard to design, prone to error and likely to be incomplete.

In order to tackle the the cold start problem in fast evolving image display ads, we propose to learn image features for online ads. We propose a new feature learning architecture to learn the most discriminative image features directly from raw pixels and user feedback in the target task. The proposed



(a) High CTR Ads

(b) Low CTR Ads

Figure 1: Sample of Display Ads

method is flexible and does not depend on human heuristic. In the situation that the image features are no longer effective, we just need to retrain our feature learning model with the latest dataset. Extensive experiments on a real world dataset with 47 billion records show that our feature learning method outperforms existing handcrafted features significantly, and it can extract discriminative and meaningful features.

The contributions of this paper are three folds.

1. We propose to extract image features in a supervised manner, in order to address the cold start problem of new image display ads in online advertising. To the best of our knowledge, this is the first paper to learn artificial ad image features for online image ads.
2. We propose a new feature learning architecture for artificial ad image feature extraction. Our proposed model learns the most discriminative image features directly from raw pixels and user feedback, and it does not depend on human heuristic. We compare our method with several state-of-the-art handcrafted image features on a large scale industrial dataset with more than 47 billion records and our method outperforms baselines significantly.
3. We gain insight into our model via correlation analysis and visualization, and we show that our model is capable of discovering discriminative and meaningful features.

The paper is organized as follows. In section 2, we describe some related work. In section 3, we formulate the click prediction problem. In section 4, we show our architecture. We present our experimental result in section 5. We conclude our work in section 6.

2 Related Work

Click prediction for online ads is the core task of online ad-network companies, and it also attracts a lot of attention in research community. Chakrabarti [Chakrabarti *et al.*, 2008] proposed to use contextual text information and click feedback data to improve click prediction. Cheng [Cheng and Cantú-Paz, 2010] and Berger [Berger *et al.*, 1996] used logistic regression model in click prediction. Kushal S. Dave [Dave and Varma, 2010] applied decision tree model in click prediction.

In order to predict CTR of new ads, many works addressed the cold start problem in various ways. Chakrabarti [Chakrabarti *et al.*, 2008] used contextual information while Kushal S. Dave [Dave and Varma, 2010] used semantically related ads. Deepak Agarwal [Agarwal *et al.*, 2010] used existing hierarchical information between ad categories to help predict CTR of new ads. However, these methods cannot be directly applied to image display ads. In the absence of enough information about the category of an ad, we will have to rely on the image feature. Due to privacy reason, some advertising systems have no access to personal information from users, and user features does not affect the comparison of different image features on the item side. In this paper, we focus on designing better general image features from the ads side, in order to tackle the cold start problem in new image ads.

Many effective handcrafted image features have been designed for various tasks. Yoo, Hun-Woo [Yoo *et al.*, 2002] used many image features to build a content based image retrieval(CBIR) system. Lowe [Lowe, 1999] proposed SIFT features for general object recognition tasks. But these handcrafted features can not be directly applied to display ads task. Cheng [Cheng *et al.*, 2012] and Javad Azimi [Azimi *et al.*, 2012] proposed to use multimedia features to predict click probability of ads in display advertising. They utilized many image features including brightness, color, contrast, sharpness, texture, interest point, saliency map, etc and they improved the state-of-the-art model significantly. However, these features are mainly fixed handcrafted features. These handcrafted features are not especially designed for click prediction, they can hardly capture the key factors for this task. And they are inflexible. In fast changing world, the important factors influencing CTR may also evolve fast, fixed handcrafted features are not flexible enough for adapting new display ads. What's worse, they rely heavily on human heuristic which is prone to error and hard to design.

Feature learning aims to learn a feature extractor from the raw inputs, such that the extracted features are effective in specific tasks. Convolutional neural network is one of the most popular feature learning architectures, which produces a hierarchy of latent features via learned filters. Krizhevsky [Krizhevsky *et al.*, 2012] used convolutional neural network to achieve record breaking performance on the image classification task on datasets of more than one million images. Zeiler [Zeiler and Fergus, 2013] found high level neurons can learned interesting and intuitive high level patterns. However, existing feature learning papers mainly focus on natural images classification. As far as we know, there is still no work on feature learning for the click prediction task in display ads. Also the existing feature learning architectures may be unsuitable for the click prediction problem.

3 Click Prediction Problem formulation

In display advertising, the ad-network runs an auction for each opportunity to display an ad to an online user, the ads with the highest effective Cost-Per-Mille(eCPM) gets the displaying opportunity. So predicting the probability that a user clicks on the ads is the core task of ad-network.

The click prediction in online advertising can be formulated as a classification problem. Each instance is an impression of an ad shown to a specific user in a specific context, along with the user's feedback on the ad. The j-th instance is formulated as $\mathcal{I}_j = \{f_j, c_j\}$, where f_j is the collection of features and c_j is the label of this instance. $f_j = \{u_j, p_j, a_j\}$, where u_j is the user side feature set, p_j is the context feature set, a_j is the advertisement side feature set. The class labels $c_j \in \{0, 1\}$ depends on the user's feedback, 0 is not clicked and 1 is clicked. We use $\mathcal{D} = \{f_j, c_j\}_{j=1}^{n}$ to denote a training set with n instances and we use $\mathcal{T} = \{f_j, c_j\}_{j=1}^{m}$ to denote a testing set with m instances. Our goal is to estimate the probability of click $p(c_j|f_j)$. Due to privacy issue, some recommender systems do not have user information, and user features does not affect the comparison of different image features on the item side. In this paper, we focus on designing better general image features a_j from advertisements side and we restrict our discussion to the case where $u_j = \emptyset$ and $p_j = \emptyset$.

We choose to use logistic regression to build our prediction model. Logistic regression model is widely used in click prediction. It is simple and easy to understand, and it can handle large number of different features. The training process can be easily extended to very large scale [Zinkevich *et al.*, 2010; Bradley *et al.*, 2011].

We predict the class label of an instance by

$$p(c_j|f_j, w) = \mathcal{G}(\Sigma_{i=1}^{d} w_i f_j^i)$$

$$\mathcal{G}(x) = \frac{1}{1 + e^x}$$

where f_j^i is the ith feature and w_i is the weight for the ith feature, d is the total number of features. The weight vector w is found by minimizing the following objective function

$$\mathcal{O}(w) = \Sigma_j^n L(w, f_j, c_j) + \frac{\lambda}{2}||w||^2$$

$$L(w, f, c) = -\log p(c|f, w)$$

where $L(x)$ is the negative log-likelihood of w given an instance $\{f, c\}$, $||w||^2$ is the L2 regularization term, λ controls the degree of L2 regularization. Minimizing the objective function equals to maximize the log-likelihood of w given the training set D.

We formulate the image feature extraction problem as follow

$$a_j = E(A_j, e)$$

A_j is the image of the jth advertisement, $E(A_j, e)$ is the function used to extract features from image, e is the feature extractor model. Handcrafted feature extractor model is fixed and do not need training. For trainable feature extractor model, we find the optimal extractor e by minimizing the following objective function

$$\bar{\mathcal{O}}(e, w) = \Sigma_j^n L(w, E(A_j, e), c_j) + \frac{\lambda}{2}||w||^2$$

In other words, we are looking for the best feature extractor e, so that we can obtain better performance on the logistic regression model with the extracted feature a_j. In this paper, the feature extractor e is a decapitated deep convolutional neural network.

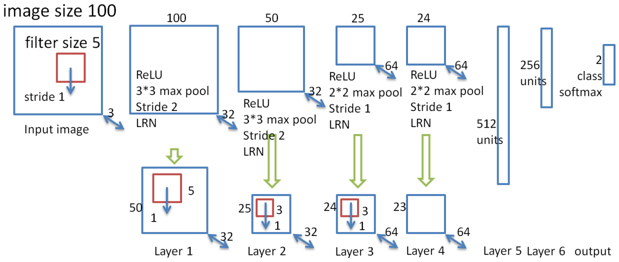


Figure 2: Architecture of our 7 layer convolutional neural network model. A 100 × 100 crop of image is used as input. The first four are convolutional layers and the remaining three are fully connected layers. The output of each convolutional layser are then passed through (i)ReLU, (ii)Maxpooling layer, (iii)Local response normalization. All local response normalization layer use $\alpha = 0.0001, \beta = 0.75$. All local response normalization have receptive filed of size 5.

4 Feature Learning Architecture

In this section, we describe the architecture of our convolutional neural network, as described in Figure 2. Below we will show some of unique characteristic of our proposed network architecture. The corresponding experiment result is in Section 5.6.

4.1 Task analysis and Architecture design

In this part we analyze the characteristics of the task. Traditional image classification tasks like ImageNet are natural images with thousands of labels. Natural images have much environmental noise and multiple objects. Learning discriminative features for thousands of labels requires a large number of common low level, mid level features. This is part of the reason why the state-of-the-art models [Krizhevsky *et al.*, 2012; Zeiler and Fergus, 2013] used a large number of filters in each convolutional layer.

In ad image click prediction case, we are mainly dealing with artificial ad images which are much simpler with less background noise and few object, as shown in Figure 1. Our network outputs "clicked" and "not clicked", so the top softmax classifier has only 2 outputs, so the common latent variables needed are far fewer than that needed to distinguish between 1000 different image classes. Over-sized model may suffer from over-fitting and have poor generalization ability, as shown in Table 4. As a result, we use much smaller number of filters in each layer.

4.2 Modeling visual element position

In this part we discuss the impact of the positions of the visual elements. Traditional image classification problem only cares about whether a visual pattern exists in the whole image. However, positions of the visual elements in an ad image are very important factors for the ad quality. Intuitively, it is easier for us to pay attention to elements in the middle of the

Table 1: Summary of Dataset Characteristics

Collections	# of Impressions	# of Clicks	# of Ads	# of Categories	# Positions
Tencent Qzone display ads	47 billion	103 million	250,000	5	5
Training set	45 billion	98 million	220,000	5	5
Testing new ads	2.4 billion	5.8 million	33,000	5	5

image. Also, different positions of the same visual element may affect the tidiness of an ad image. In our model, we model the position factor better, by using a larger output feature map for the convolutional layers. Experimental results in Table 4 also show that using small feature map will deteriorate model performance.

4.3 Processing large dataset on a single machine

In this part we introduce the techniques we used to speed up our training process. The dataset we used contains 47 billion instances and each instance corresponds to an impression of an ad. Obviously, one machine cannot handle such a large dataset in a normal way. Since we do not have user feature, we can merge all instances with the same adid and the same ad position together into an aggregated instance. An aggregate instance has a 2 dimension label, the first dimension records the total number of “not clicked” instances and the second dimension records the total number of “clicked” instances. For example, an ad with 10 “not clicked” instances and 2 “clicked” instances have label $< 10, 2 >$. We implement a neural network which is efficient for these 2 dimension label.

4.4 Reduce over-fitting

In this part, we describe the techniques we used to reduce over-fitting.

Data augmentation [Ciresan *et al.*, 2012; Simard *et al.*, 2003] enlarge the dataset by producing easy label-preserving transformation to the images. We first resize shortest edges of each image to 128 pixels in length. Then in each pass, we randomly crop each image and produce a 100×100 subimage in the training set. In testing phase, we use 10 crops like [Krizhevsky *et al.*, 2012].

Drop out [Hinton *et al.*, 2012] can alleviate the over-fitting caused by highly correlated features. It randomly discards some of the activated neuron in the output. We apply drop out to the fully connected layers in our architecture.

Local response normalization(LRN) [Krizhevsky *et al.*, 2012] is a technique to normalize brightness of the input image. We find that applying local response normalization after the high level convolutional layers can further improve performance. We also used ReLU [Nair and Hinton, 2010] as activation function which do not saturate and produce sparse activation. Experimental results are in Table 4.

4.5 Training details

In this part we introduce the details of the training. We modified Caffe [Jia, 2013] to train our ad image feature extractor network. The objective is to minimize the soft-max loss. The model was trained using momentum and Nesterov’s Accelerated Gradient [Sutskever *et al.*, 2013]. We used a batch size of 256, weight decay of 0.0005. In order to speed up the convergence, we double the batch size after every 5000 iterations.

Learning rate is adjusted dynamically according to a heuristic [Lan, 2012; Hu *et al.*, 2009] $\mathbf{L}_i = (\bar{\mathbf{L}} + \gamma * \mathbf{i}^{\mathbf{p}^{-1}})^{-1}$ where $\mathbf{L}_i$ is the learning rate for i-th iteration, γ and $\mathbf{p}$ are two hyper parameters to control the evolving speed. The basic learning rate $\bar{\mathbf{L}}$ is set to be 0.01, γ is set to be 0.0001 and $\mathbf{p}$ is set to be 1.5.

Momentum is adjusted dynamically according to a heuristic, $\mathbf{M}_i = \min(\bar{\mathbf{M}}, (1 - 2^{-1-\log_2^{\frac{i}{\hat{\mathbf{M}}}+1}}))$ where $\mathbf{M}_i$ is the momentum for i-th iteration, $\bar{\mathbf{M}}$ is the max momentum, $\hat{\mathbf{M}}$ is the number of iterations before a momentum update. We set the max momentum to be 0.9, $\hat{\mathbf{M}}$ to be 500.

4.6 Trade off between efficiency and performance

The model converges after 60000 iterations, which are about 60 epochs. Training the feature extractor takes about 2 days on a NVIDIA TESLA M2090 6GB GPU. Training our proposed feature extractor model is comparative slower than directly extracting predefined handcrafted features. However, it requires no human knowledge and might save a lot of human effort in designing whole new handcrafted image features for a specific task. The proposed method is suitable for extracting image feature for fast evolving CTR prediction tasks with little or no human knowledge.

5 Experiment

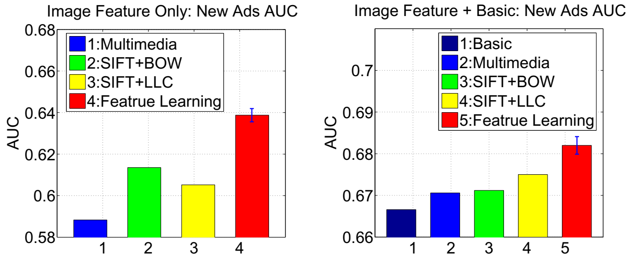
In this section, we experimentally verify that our feature learning method, compared with methods using current state-of-the-art handcrafted image features on a large scale industrial dataset. First we compare our feature learning method with handcrafted features on identifying potential popular ads images. Then we compare our feature learning method with handcrafted image features in the presence of some other existing ad features. Finally we then gain insight into the feature extractor model by analysing correlation with unseen ad categories and visualizing discriminative areas.

5.1 Experiment setting

In this section we first introduce the dataset, then we introduce the baseline image features, finally we describe the full steps of the experiments. For evaluation metrics, we introduce area under ROC curve(AUC) to measure how accurate the predicted result matches the ground truth.

Dataset

Our dataset is sampled from the Tencent online display advertising system for a period of 19 days, with approximately 47 billion records. Each record is an impression of an ads



(a) Image Feature Only

(b) Image Feature + Basic

Figure 3: CTR prediction Result

Table 2: AUC obtained in different experiment setting

Algorithm	Experiment Setting	
	Image Feature Only	Image Feature + Basic Features
Basic Features	-	0.6666
Multimedia	0.5883	0.6706
SIFT+BOW	0.6135	0.6712
SIFT+LLC	0.6052	0.6750
Feature Learning	**0.6387 ± 0.0032**	**0.6820 ± 0.0021**

and is labeled as “clicked” or “not clicked”, which is calculated with the user click log. We sampled our dataset from the records of 5 popular advertisement categories from 5 displaying positions in Tencent Qzone web pages. Note that the same ad may appear in different displaying positions. There are approximately 250,000 unique display ads in our dataset. We show some statistics of the dataset in Table 1 and some sample ad images in Figure 1. The records from the first 15 days are used as training set, the records from the last 4 days are used as testing set. The training set have 45 billion records on 220,000 ads. The test set have 2.4 billion records on about 33,000 different new ads that did not appear in the training set. We will report testing result on new ads in testing set.

Baselines

We compare our feature learning method with 2 kinds of handcrafted feature baselines. (1) Multimedia Features [Cheng *et al.*, 2012; Geng *et al.*, 2011], including Lighting, Color, Sharpness and Blur, Subject Quality, Rule of Third, Simplicity, Visual weight, Dynamics, Color Emotion, which is the set of mainstream handcrafted features used in image search and display advertising for the cold start problem. There are a total of 53 multimedia features. (2) Scale-Invariant feature transformation [Lowe, 1999] with Bag of Words [Sivic and Zisserman, 2009] (SIFT+BOW). Scale-Invariant feature transformation with Locality-constrained Linear Coding [Wang *et al.*, 2010] (SIFT+LLC). Both SIFT+BOW and SIFT+LLC features have 256 dimensions. Due to privacy reason, we do not use personal information from users and it will not affect the comparison of different image features. In this paper, we focus on learning better image features for advertisements.

Table 3: Basic advertisement features in baseline model for click prediction

Feature	Number	Feature description
Ad ID	250,000	Unique ID for each ads
Ad Category	5	Indicate the category of this ads
Ad Position	5	The displaying position ID of ads

Experiment Steps

1. Image Feature Extraction is the first step of the experiment. For Multimedia image features and SIFT+BOW image features, we directly extract all features for each of the ad images, with the help of with OpenCV [Bradski,]. For SIFT+LLC, we use the code provided in the original paper. For our feature learning method, we first train the feature extractor. Then we use the normalized output of this feature extractor(decapitated neural network) as image features. We repeat the feature learning process for 10 times.

2. Model Training. For Section 5.2, we use only image features to predict CTR of an ad. In this case, an instance contains the required image features and the label. For Section 5.3, we combine image features with the adid, ad category and ad position of the impression. We train a logistic regression model with liblinear [Fan *et al.*, 2008] on the training set.

3. Testing. We predict click probability of every instance in new ads set, then we calculate and report the AUC of new ads. For our non-deterministic feature extractor, we report the AUC mean and AUC std of 10 runs.

5.2 Image features only comparison

In this part we compare different kinds of image features by building prediction models using only image features. We have 4 sets of image features, namely “Multimedia”, “SIFT+BOW”, “SIFT+LLC” and “Feature Learning”. We train a logistic regression model using each set of these features. The results are shown in Figure 3 and Table 2.

We find that our feature learning method outperforms both multimedia image features and SIFT features by as much as 4.1% in predicting new ads. This shows that our feature learning method is more suitable for click prediction, and generalizes better to unseen ad images.

5.3 Combining image feature with basic features

In this part we compare different kinds of image features by building prediction models using both image features and basic ads features. The basic ads features includes Ad ID, Ad Category, Ad Position, as listed in Table 3. For each of “Multimedia”, “SIFT+BOW”, “SIFT+LLC” and “Feature Learning”, we build logistic regression models along with basic ads features. Besides the above baselines, we build a model “Basic” using only the basic ad features. The results are shown in Figure 3 and Table 2.

We find that “Feature Learning” outperforms both kinds of handcrafted features by a large margin. It again proves that our feature learning method is very suitable and effective in predicting new ads, even in the presence of other ad features.

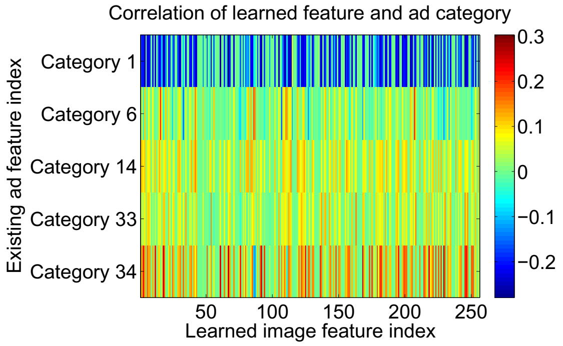


Figure 4: Correlation of learned image features and ad category(best viewed in color)

This experiment shows that we can combine our feature learning method with other ad features and further improve the performance of an existing online advertising system.

5.4 Discriminative image features: ad categories

In this section, we will gain insight into our model by analysing correlation between output of our model and ground truth categories of ads. Note that we did not use the categories of ads to train our feature extractor.

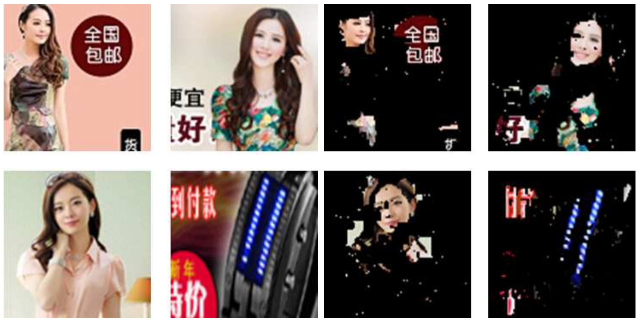
The pearson correlation between every learned feature and every dummy ad category are presented in Fig 4. We find that the resulting pearson correlation ranged from -0.27 to 0.3. Some learned image features are correlated to category 1 and 34. It seems that our feature extractor has learned to discriminate between some ad categories. The reason might be different ad categories naturally have different CTR. For example, women dress ads have high CTR than other categories. In this dataset, ads in category 1 and 34 do have different mean CTR compared with mean CTR of all ads. Our model has learn to distinguish between different categories because these categories are useful for predicting CTR of ad images. The experiment result shows that our proposed feature learning architecture can discover discriminative features in image ads.

Table 4: AUC of different feature learning architectures

Chosen	No lrn	Samll	Fat	Tall	Short
0.6387	0.6285	0.6244	0.6309	0.6222	0.6224

5.5 Visualizing discriminative area

In this section we gain insight into what the model have learn by visualizing the discriminative areas of some ad images. We follow the method used in [Simonyan *et al.*, 2013]. Some typical discriminative areas are shown in Fig 5. Although there are some noise, we can still see some meaningful indicators of high click through rate. Human face seems to be an indicator, which imply the model learned that ads with human face have higher click through rate. Some character used in promotion seems to be another indicator of high click through rate. This experiment shows that our feature extractor can learn to extract meaningful features.



(a) Ads (b) Discriminative area

Figure 5: Discriminative areas of ad images

5.6 Architecture selection

In this section we conduct some experiments to show the impact of various factors during architecture selection process in Section 4. The result is listed in Table 4. “Fat” has twice as many outputs in each layer, “Tall” model have 5 convlolutional layers and “short” model have 3 convolutional layers. All of them have inferior performance. It shows the importance of an appriориate model capacity according to the nature of the problem and the training dataset. “Small” have half feature map size in the last convolutional layer and lower performance, which shows that position of visual element is very important factor in display ads. “No lrn” has no LRN after the 3rd and 4th convolutional layer and it is no as good as “Chosen” architecture. It shows that normalization layer further reduce the number of extreme value in high level feature map, which helps the training of higher level layers.

6 Conclusion

We propose to extract image features in a supervised manner, in order to address the cold start problem of new image display ads in online advertising. To the best of our knowledge, this is the first paper to learn artificial ad image features for online image ads. We propose a new feature learning architecture for artificial ad image feature extraction. Our proposed model learns the most discriminative image features directly from raw pixels and user feedback in the target task. And the proposed method is flexible and does not depend on human heuristic. We evaluate and compare our method with several state-of-the-art handcrafted image features including multimedia features and SIFT features on a large scale industrial dataset with more than 47 billion records, where our feature learning method outperforms baselines significantly. We gain insight into our model via correlation analysis and visualization, and we show that our model is capable of discovering discriminative and meaningful features.

Acknowledgement

We thank the support of China National Fundamental Research Project (973 Program) under Project #2014CB340304. We also thank the support of the ITF project ITS/191/13FX.

References

[Agarwal *et al.*, 2010] Deepak Agarwal, Rahul Agrawal, Rajiv Khanna, and Nagaraj Kota. 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*, pages 213–222. ACM, 2010.

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

[Berger *et al.*, 1996] Adam L Berger, Vincent J Della Pietra, and Stephen A Della Pietra. A maximum entropy approach to natural language processing. *Computational linguistics*, 22(1):39–71, 1996.

[Bradley *et al.*, 2011] Joseph K Bradley, Aapo Kyrola, Danny Bickson, and Carlos Guestrin. Parallel coordinate descent for l1-regularized loss minimization. *arXiv preprint arXiv:1105.5379*, 2011.

[Bradski,] G. Bradski. *Dr. Dobb's Journal of Software Tools*.

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

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

[Cheng *et al.*, 2012] Haibin Cheng, Roelof van Zwol, Javad Azimi, Eren Manavoglu, Ruofei Zhang, Yang Zhou, and Vidhya Navalpakkam. Multimedia features for click prediction of new ads in display advertising. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 777–785. ACM, 2012.

[Ciresan *et al.*, 2012] Dan Ciresan, Ueli Meier, and Jürgen Schmidhuber. Multi-column deep neural networks for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3642–3649. IEEE, 2012.

[Dave and Varma, 2010] Kushal S Dave and Vasudeva Varma. Learning the click-through rate for rare/new ads from similar ads. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 897–898. ACM, 2010.

[Fan *et al.*, 2008] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[Geng *et al.*, 2011] Bo Geng, Linjun Yang, Chao Xu, Xian-Sheng Hua, and Shipeng Li. The role of attractiveness in web image search. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 63–72. ACM, 2011.

[Hinton *et al.*, 2012] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[Hu *et al.*, 2009] Chonghai Hu, Weike Pan, and James T Kwok. Accelerated gradient methods for stochastic optimization and online learning. In *Advances in Neural Information Processing Systems*, pages 781–789, 2009.

[Jia, 2013] Yangqing Jia. Caffe: An open source convolutional architecture for fast feature embedding. `http://caffe.berkeleyvision.org/`, 2013.

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[Lan, 2012] Guanghui Lan. An optimal method for stochastic composite optimization. *Mathematical Programming*, 133(1-2):365–397, 2012.

[Lowe, 1999] David G Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee, 1999.

[Nair and Hinton, 2010] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814, 2010.

[Simard *et al.*, 2003] Patrice Y Simard, Dave Steinkraus, and John C Platt. Best practices for convolutional neural networks applied to visual document analysis. In *2013 12th International Conference on Document Analysis and Recognition*, volume 2, pages 958–958. IEEE Computer Society, 2003.

[Simonyan *et al.*, 2013] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[Sivic and Zisserman, 2009] Josef Sivic and Andrew Zisserman. Efficient visual search of videos cast as text retrieval. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(4):591–606, 2009.

[Sutskever *et al.*, 2013] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 1139–1147, 2013.

[Wang *et al.*, 2010] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas Huang, and Yihong Gong. Locality-constrained linear coding for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3360–3367. IEEE, 2010.

[Yoo *et al.*, 2002] Hun-Woo Yoo, Dong-Sik Jang, Seh-Hwan Jung, Jin-Hyung Park, and Kwang-Seop Song. Visual information retrieval system via content-based approach. *Pattern Recognition*, 35(3):749–769, 2002.

[Zeiler and Fergus, 2013] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional neural networks. *arXiv preprint arXiv:1311.2901*, 2013.

[Zinkevich *et al.*, 2010] Martin Zinkevich, Markus Weimer, Lihong Li, and Alex J Smola. Parallelized stochastic gradient descent. In *Advances in Neural Information Processing Systems*, pages 2595–2603, 2010.