Metadata of the chapter that will be visualized in SpringerLink

Book Title	Machine Learning and Knowledge Discovery in Databases	
Series Title		
Chapter Title	Image-to-Markup Generation via Paired Adversarial Learning	
Copyright Year	2019	
Copyright HolderName	Springer Nature Switzerland AG	
Corresponding Author	Family Name	**Wu**
	Particle	
	Given Name	**Jin-Wen**
	Prefix	
	Suffix	
	Role	
	Division	NLPR, Institute of Automation
	Organization	Chinese Academy of Sciences
	Address	Beijing, People's Republic of China
	Division	
	Organization	University of Chinese Academy of Sciences
	Address	Beijing, People's Republic of China
	Email	jinwen.wu@nlpr.ia.ac.cn
Author	Family Name	**Yin**
	Particle	
	Given Name	**Fei**
	Prefix	
	Suffix	
	Role	
	Division	NLPR, Institute of Automation
	Organization	Chinese Academy of Sciences
	Address	Beijing, People's Republic of China
	Email	fyin@nlpr.ia.ac.cn
Author	Family Name	**Zhang**
	Particle	
	Given Name	**Yan-Ming**
	Prefix	
	Suffix	
	Role	
	Division	NLPR, Institute of Automation
	Organization	Chinese Academy of Sciences
	Address	Beijing, People's Republic of China
	Email	ymzhang@nlpr.ia.ac.cn
Author	Family Name	**Zhang**
	Particle	

	Given Name	**Xu-Yao**
	Prefix	
	Suffix	
	Role	
	Division	NLPR, Institute of Automation
	Organization	Chinese Academy of Sciences
	Address	Beijing, People's Republic of China
	Email	xyz@nlpr.ia.ac.cn
Author	Family Name	**Liu**
	Particle	
	Given Name	**Cheng-Lin**
	Prefix	
	Suffix	
	Role	
	Division	NLPR, Institute of Automation
	Organization	Chinese Academy of Sciences
	Address	Beijing, People's Republic of China
	Division	
	Organization	University of Chinese Academy of Sciences
	Address	Beijing, People's Republic of China
	Division	
	Organization	CAS Center for Excellence of Brain Science and Intelligence Technology
	Address	Beijing, People's Republic of China
	Email	liucl@nlpr.ia.ac.cn

Abstract	Motivated by the fact that humans can grasp semantic-invariant features shared by the same category while attention-based models focus mainly on discriminative features of each object, we propose a scalable paired adversarial learning (PAL) method for image-to-markup generation. PAL can incorporate the prior knowledge of standard templates to guide the attention-based model for discovering semantic-invariant features when the model pays attention to regions of interest. Furthermore, we also extend the convolutional attention mechanism to speed up the image-to-markup parsing process while achieving competitive performance compared with recurrent attention models. We evaluate the proposed method in the scenario of handwritten-image-to-LaTeX generation, i.e., converting handwritten mathematical expressions to LaTeX. Experimental results show that our method can significantly improve the generalization performance over standard attention-based encoder-decoder models.

Image-to-Markup Generation via Paired Adversarial Learning

Jin-Wen Wu[1,2(✉)], Fei Yin[1], Yan-Ming Zhang[1], Xu-Yao Zhang[1], and Cheng-Lin Liu[1,2,3]

[1] NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing, People's Republic of China
{jinwen.wu,fyin,ymzhang,xyz,liucl}@nlpr.ia.ac.cn
[2] University of Chinese Academy of Sciences, Beijing, People's Republic of China
[3] CAS Center for Excellence of Brain Science and Intelligence Technology, Beijing, People's Republic of China

Abstract. Motivated by the fact that humans can grasp semantic-invariant features shared by the same category while attention-based models focus mainly on discriminative features of each object, we propose a scalable paired adversarial learning (PAL) method for image-to-markup generation. PAL can incorporate the prior knowledge of standard templates to guide the attention-based model for discovering semantic-invariant features when the model pays attention to regions of interest. Furthermore, we also extend the convolutional attention mechanism to speed up the image-to-markup parsing process while achieving competitive performance compared with recurrent attention models. We evaluate the proposed method in the scenario of handwritten-image-to-LaTeX generation, i.e., converting handwritten mathematical expressions to LaTeX. Experimental results show that our method can significantly improve the generalization performance over standard attention-based encoder-decoder models.

Keywords: Paired adversarial learning · Semantic-invariant features
Convolutional attention · Handwritten-image-to-LaTeX generation

1 Introduction

The image-to-markup problem has attracted interest of researchers from 1960s [2]. The main target of the research is recognition for the printed/handwritten mathematical expressions (MEs). Different from typical sequence-to-sequence tasks such as machine translation [4] and speech recognition [8], image-to-markup generation converts the two-dimensional (2D) images into sequences of structural presentational languages. More specifically, it has to scan the two dimensional grids to generate markup of the symbols and the implicit spatial operators, such as subscript and fractions. Image-to-markup generation is also different from other image-to-sequence tasks such as image captioning [7,24] and text string
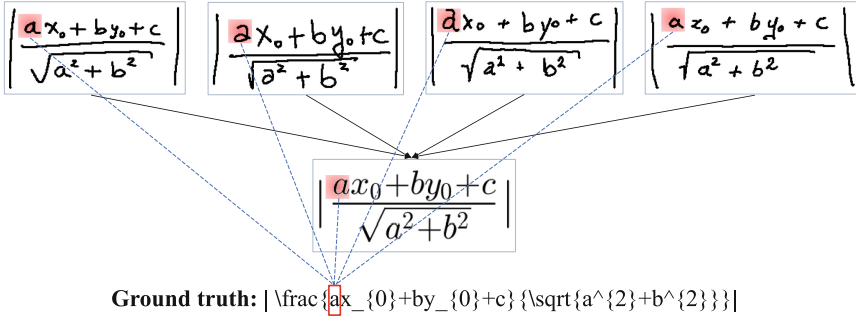
Ground truth: $| \frac{ax_{0}+by_{0}+c}{\sqrt{a^{2}+b^{2}}} |$

Fig. 1. MEs written by different people (top) and their standard printed template (center). These images have the same ground-truth markup (bottom). The red cells indicate the attention at same symbol a. Same symbols might be written in very different styles while share invariant features that represent the same semantic meaning. (Color figure online)

recognition in optical character recognition [30], in that input images in image-to-markup problem contain much more structural information and spatial relations than general images in computer vision.

Traditional approaches for the image-to-markup problem use handcrafted grammars to handle symbol segmentation, symbol recognition and structural analysis. Although grammar-driven approaches [1,3,23,26] can achieve high performance in practice, they require a large amount of manual work to develop grammatical rules. Furthermore, grammar-driven structural analysis is also highly computationally demanding.

Recently, methods based on deep neural networks have been proposed for image-to-markup generation and achieved great success [10,11,20,28,29]. For example, the model WYGIWYS extended the attention-based encoder-decoder architecture to image-to-markup problem [11]. It encodes printed MEs images with a multi-layer CNN and a bidirectional LSTM and employs a LSTM as the recurrent attention based decoder to generate the target LaTeX format markup. To speedup the method, authors of [10] improved the original WYGIWYS with coarse-to-fine attention and performed experiments on synthetic handwritten MEs. These studies show the data-driven attention-based models can be as effective as the grammar-based approaches while exploiting no prior knowledge of the language.

For the image-to-markup problem, it is especially important to ensure the translation of each local region of the input image. Motivated by this observation, the model WAP [29] records the history of attention at all local regions for improving the coverage of translation. An improved version of WAP uses deep gated recurrent unit (GRU) to encode the online trajectory information of handwritten MEs [28], and has achieved the state-of-the-art performance using an ensemble of five models.

Despite the progresses achieved so far, handwritten-image-to-markup generation is still a very challenging task due to the highly variable handwriting styles compared with printed images, see Fig. 1. On the other hand, well-annotated

handwritten MEs are rather scarce. For example, the currently largest public database of handwritten mathematical expression recognition, the Competition on Recognition of Online Handwritten Mathematical Expressions (CROHME) database, contains only $8,836$ MEs. In order to alleviate the contradiction between the limited training data and the great writing-style variation, it is common to augment the training dataset by distorting the input images [20].

To overcome the scarcity of annotated training data in handwritten-image-to-markup generation, we propose an attention-based model with paired adversarial learning for learning semantic-invariant features. The main contributions of this paper are as follows: (1) we present a scalable paired adversarial learning (PAL) method incorporating the prior knowledge of standard templates to guide the attention-based model to learn intrinsic semantic-invariant features; (2) we use a fully convolutional attention based decoder to speed up the image-to-markup decoding without losing accuracy; (3) we introduce a novel multi-directional transition layer that can be easily extended to other deep convolutional networks for accessing 2D contextual information.

2   Background

Before describing our proposed method, we briefly review the generative adversarial network (GAN) [14] in Sect. 2.1 and the convolutional attention (ConvAttention) model [13] proposed for sequence-to-sequence learning in Sect. 2.2.

2.1   Generative Adversarial Network

GAN is a well-known adversarial learning method originally presented for generative learning by Goodfellow et al. [14]. It generally consists of a generator G and a discriminator D, which are trained with conflicting objectives:

$$\min_{G} \max_{D} V(G, D) = E_{\mathbf{x} \sim p_{data}(\mathbf{x})}[\log D(\mathbf{x})] + E_{\mathbf{z} \sim p_{\mathbf{z}}(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))] \quad (1)$$

where $\mathbf{x}$ denotes the target real sample, $\mathbf{z}$ is the input noise and $D(\mathbf{x})$ is the probability that the sample is real. G tries to forge real samples to confuse D while D tries to distinguish fake samples from real ones. Adversarial learning method does force both G and D to improve and has been proven effective in producing highly realistic samples [6,12,25].

Recently, the idea of adversarial learning in GAN has been applied to the image-to-image translation task and demonstrates very encouraging results [18,31]. It is interesting to observe that D successfully guides G to learn the style information from the two domains and realize style transfer from the source domain to the target domain. Another work related to our proposal is the domain adaptation with GAN [5], in which G is guided by D to find a *domain-invariant representation* to represent two domains with different distributions. Inspired by these works, we design an attention-based model to grasp semantic-invariant features from symbols with different writing-styles under the adversarial learning framework.

2.2   Convolutional Attention

Though recurrent attention performs well in dealing with sequential problems, it is still time consuming due to its sequential structure. In this section, we briefly introduce the fully convolutional attention (Conv-Attention) [13], which is proposed for machine translation and shows competitive performance while being more efficient.

Suppose the input sequence is $\mathbf{w} = (w_1, \cdots, w_N)$. $\mathbf{w}$ is then embedded in a distributional space as $\mathbf{x} = (x_1, \cdots, x_N), x_j \in \mathbb{R}^D$ and the absolute position of input elements is embedded as $\mathbf{p} = (p_1, \cdots, p_N), p_j \in \mathbb{R}^D$. The input of the encoder is finally represented as $\mathbf{e} = (x_1 + p_1, \cdots, x_N + p_N)$ to guarantee the model's sense of order and the output sequence of the encoder is $\mathbf{f} = (f_1, \cdots, f_N), f_j \in \mathbb{R}^D$. This process has been also applied to the output elements already generated by the decoder.

The encoder and the Conv-Attention based decoder share a simple block structure. Each block (or referred to layer) contains a one dimensional convolution and a subsequent non-linearity, and computes the output states with a fixed number of input elements. Each convolution kernel of the decoder blocks is parameterized as $W \in \mathbb{R}^{2D \times kD}$ with a base $b_w \in \mathbb{R}^{2D}$, where k is the kernel width and D is the channel dimension of the input features. This convolution kernel maps k concatenated input elements which are embedded in D dimensions to a single output $o_j \in \mathbb{R}^{2D}$. The following non-linearity of one dimensional convolution is chosen as gated linear units (GLU) [9] that implements a gating mechanism over each output element $o_j = [o_{j1} \; o_{j2}] \in \mathbb{R}^{2D}$:

$$GLU(o_j) = o_{j1} \odot \sigma(o_{j2}) \tag{2}$$

where the $\odot$ denotes the point-wise multiplication and gates $\sigma(o_{j2})$ determine which parts of o_{j1} are relevant. The output $GLU(o_j) \in \mathbb{R}^D$ is half the channel dimension of the input o_j.

Conv-Attention uses a separate attention mechanism for each decoder block. It first computes the state summary s_i^l with previous target embedding $\mathbf{t} = (t_1, \cdots, t_T), t_i \in \mathbb{R}^D$ and current hidden state of the l-th block $\mathbf{h}^l = (h_1^l, \cdots, h_T^l), h_i^l \in \mathbb{R}^D$ as:

$$s_i^l = W_s^l h_i^l + b_s^l + t_i \tag{3}$$

Next, attention score $\alpha_{i,j}^l$ of state i and source element j is calculated via a dot-product between f_j of the feature sequence $\mathbf{f}$ and state summary s_i^l:

$$\alpha_{i,j}^l = \frac{\exp(s_i^l \cdot f_j)}{\sum_{w=1}^N \exp(s_i^l \cdot f_w)} \tag{4}$$

After the weights $\alpha_{i,j}^l$ have been computed, the context vector is calculated as:

$$c_i^l = \sum_{j=1}^N \alpha_{i,j}^l (f_j + e_j) \tag{5}$$

Then, the context vector is simply added to the corresponding hidden feature h_i^l. This operation can be considered as attention with multiple *hops* [27], which improves the model's ability to access more attention history. Furthermore, to improve the information flow between blocks, residual connections are added from input to output as ResNet in [16].

3   Paired Adversarial Learning

The motivation of our work is to make the model learn semantic-invariant features of patterns to conquer the difficulties caused by the writing-style variation and the small sample size. Roughly speaking, for each handwritten image in the training set, we first generate its printed image template by compiling the LaTeX format label with a general LaTeX editor. Then, we force the attention-based encode-decoder model to extract similar features for both the handwritten image and its printed image template, which is implemented under the adversarial learning framework.
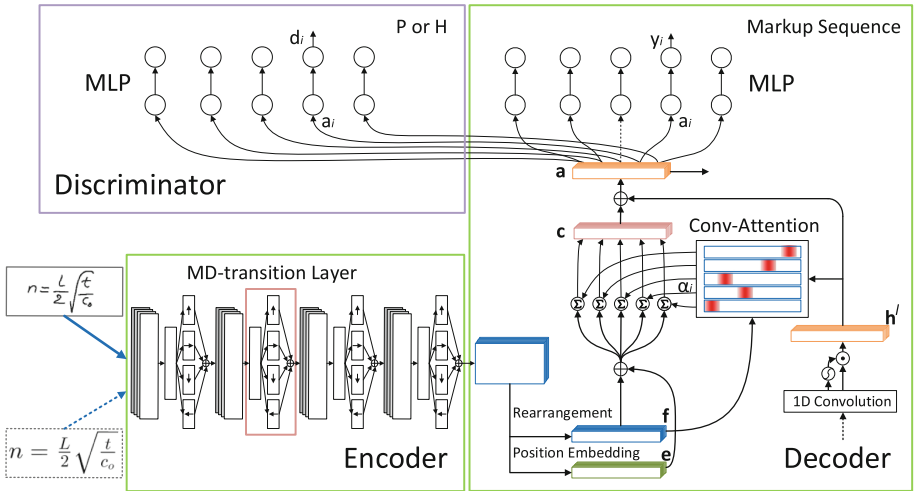

Fig. 2. Architecture of the paired adversarial learning (PAL). When training, each handwritten image is input with its paired printed template (bottom left). The encoder-decoder model and the discriminator are trained alternatively. We use Conv-Attention based decoder here to speed up the image-to-markup decoding. Theoretically, the attention model can also be substituted with any standard recurrent attention.

The proposed model consists of three parts (see Fig. 2): an encoder that extracts features from images, a decoder that parses the sequence of features outputted by the encoder and generates the markup, and a discriminator trained against the encoder-decoder model to force it to learn the semantic-invariant feature of each pattern. In the following subsections, we will sequentially introduce the encoder, the decoder, the learning objective of the proposed paired adversarial learning method, and the training algorithm.

3.1   Multi-directional Encoder

The basic architecture of the encoder in this work is adapted from the fully convolutional network (FCN) model in [29]. The difference is that we introduce a novel layer, named MD-transition layer, equipped after each convolutional block of the deep FCN model. We utilize the multi-dimensional long short-term memory (MDLSTM) [15] to improve FCN's ability to access the 2D contextual information and apply a pooling layer [21] before the MDLSTM layer to improve computation efficiency. We refer to this architecture as MD-transition layer.

MDLSTM employs LSTM layers in up, down, left and right directions. Different LSTM layers are executed in parallel to improve the computation efficiency. After the LSTM layers, we collapse the feature maps of different directions by simply summing them up. The LSTM layers in the horizontal and vertical directions are calculated as:

$$(y_{i,j}, c_{i,j}) = LSTM(x_{i,j}, y_{i\pm1,j}, y_{i,j\pm1}, c_{i\pm1,j}, c_{i,j\pm1}) \tag{6}$$

where y and c denote the output feature vector and inner state of the cell, respectively, and $x_{i,j}$ denotes the input vector of the feature map at position (i,j). The $LSTM$ denotes the mapping function of general LSTM networks which process the input sequence over space or time. With this set up, the subsequent FCN block is enabled to access more past and future contextual information in both horizontal and vertical directions.

3.2   Decoder with Convolutional Attention

We extend the decoder with Conv-Attention [13] to generate markup for the images and speed up the decoding. Different from machine translation, image-to-markup generation is a 2D-to-sequence problem. Since the outputs of the encoder are in the form of 2D feature maps rather than feature sequences, we have to propose a conversion method that preserves feature information as much as possible.

Suppose the output feature map of the multi-directional feature extractor sizes $H \times W \times D$. We split the feature map by columns and then concatenate them to get the feature sequence $\mathbf{f} = (f_1, \cdots, f_N), f_j \in \mathbb{R}^D, N = H \times W$. Then, in order to guarantee the position information during conversion, we add $\mathbf{f}$ with the embedding of the absolute position and get embedded feature sequence $\mathbf{e} = (e_1, \cdots, e_N), e_j \in \mathbb{R}^D$. Here, $\mathbf{e}$ is not the input of the encoder as the original work. After getting $\mathbf{f}$ and $\mathbf{e}$, we compute the context vectors as Sect. 2.2. With rearrangement of the feature map and position embedding, Conv-Attention can be successfully applied to image-to-markup generation. In this study, blocks number l of the Conv-Attention is set to 3. Via the multi-step attention, the model is enabled to access more attention history, thereby improving the ability of consistent tracking with its attention.

3.3   Paired Adversarial Learning

Adversarial learning for image-to-markup task are more complex than image generation, since *mismatch* between two sequences of feature vectors can easily cause the discriminator converging to irrelevant features, and thus lose the ability of guiding the encoder-decoder model to learn the semantic-invariant features. To settle this problem, first, we pair each handwritten image with its same-size printed template to ensure that the length of the two feature sequences are same. Second, since the labels of the paired images are same, the feature vectors at the same position of these two feature sequences are forced to be extracted from related regions with the attention mechanism.

Specifically, let $\mathbf{a}(x, \phi_E) = (a_1, \cdots, a_T), a_i \in \mathbb{R}^D$ denote the feature sequence at the last feature extraction layer of the decoder. Here, x is the input handwritten image x_h or its paired printed template x_p and ϕ_E denotes the parameters of the encoder-decoder model. Our model learns the semantic-invariant features with the guide of a discriminator D which judges whether a feature vector comes from the handwritten images or the printed templates. Let $D(a_i(x, \phi_E), \phi_D)$ represent the probability that feature vector a_i comes from a printed image and ϕ_D denotes the parameters of D. The objective function is defined as:

$$\mathcal{L}_D = E_{(x_h, x_p) \sim X}[E_{a_i(x_p, \phi_E) \sim \mathbf{a}}[\log D(a_i(x_p, \phi_E), \phi_D)] + \\ E_{a_i(x_h, \phi_E) \sim \mathbf{a}}[\log(1 - D(a_i(x_h, \phi_E), \phi_D)]] \tag{7}$$

where $X = \{(x_h, x_p)\}$ is the set of paired training images. D is optimized to maximize the probability of assigning correct labels to the extracted featu res by maximizing $\mathcal{L}_D$. On the contrary, the encoder-decoder model is trained to learn semantic-invariant features to confuse D by minimizing $\mathcal{L}_D$.

Moreover, the primary goal of the encoder-decoder model is to extract discriminative features and generate the correct markup. Thus, the decoder has to convert the feature sequence to the markup by a classification layer as:

$$p(y_{a_i} = y_i | x; \phi_E) = \frac{\exp(C(y_{a_i} = y_i | \mathbf{a}(x, \phi_E)))}{\sum_{l=1}^{L} \exp(C(y_{a_i} = l | \mathbf{a}(x, \phi_E)))} \tag{8}$$

Here, $y_i \in Y = \{1, \ldots, L\}, L$ denotes the total class number of the label set, y_{a_i} is the prediction of feature vector a_i in the feature sequence $\mathbf{a}(x, \phi_E)$.

Ideally, features extracted from both the printed and handwritten images should be classified correctly with high probabilities. The cross-entropy objective function for classifying the features learned from printed images is defined as:

$$\mathcal{L}_{C_p} = -E_{x_p \sim X_p}[\sum_{i=1}^{T} \log p(y_{a_i} = y_i | x_p; \phi_E)] \tag{9}$$

where $X_p = \{x_p\}$ is the set of printed image templates. Similarly, the loss function for classifying the features learned from handwritten images is defined as:

$$\mathcal{L}_{C_h} = -E_{x_h \sim X_h}[\sum_{i=1}^{T} \log p(y_{a_i} = y_i | x_h; \phi_E)] \tag{10}$$

where $X_h = \{x_h\}$ is the set of handwritten images.

In summary, we train the attention-based encoder-decoder model by minimizing the loss function of:

$$\mathcal{L}_E = \mathcal{L}_{C_p} + \mathcal{L}_{C_h} + \lambda \mathcal{L}_D \tag{11}$$

λ is a hyper-parameter that controls the tradeoff between the discriminative features and the semantic-invariant features. When $\lambda = 0$, the method is a general attention-based encoder-decoder model trained on the paired samples. When λ increases, the method will focus more on learning the semantic-invariant features and extract less discriminative features for the classification layer to generate the predictions.

3.4   Training Procedure

The encoder-decoder model and discriminator D are trained jointly with the paired adversarial learning algorithm. D is optimized with the objective of distinguishing the sequences of feature vectors extracted from the handwritten images and the printed templates. Contrarily, the encoder-decoder model is optimized to extract more sophisticated semantic-invariant features to fool D. Meanwhile, the encoder-decoder model is trained to maximize the probability of ground-truth markup symbols of the input images. The importance of these two objective function is balanced via the hyper-parameter λ.

See details in Algorithm 1. We sample minibatch of the paired samples to train the encoder-decoder model and D for every training cycle. The encoder-decoder model is trained one time first, and D is trained k times then. The parameters of these models are updated by adaptive moment estimation (Adam). Specifically, we update the parameters for the encode-decoder model as:

$$\phi_E \leftarrow \phi_E - Adam(\frac{\partial(\mathcal{L}_{C_p} + \mathcal{L}_{C_h} + \lambda\mathcal{L}_D)}{\partial\phi_E}, \eta_E) \tag{12}$$

And for the discriminator by:

$$\phi_D \leftarrow \phi_D + Adam(\frac{\partial\mathcal{L}_D}{\partial\phi_D}, \eta_D) \tag{13}$$

Here, the *Adam* is the function to compute the updated value of the adaptive moment estimation with the gradient and learning rate, η_E denotes the learning rate for the encoder-decoder model and η_D denotes the learning rate of the discriminator. See more details in Algorithm 1.

Algorithm 1. The Paired Adversarial Learning Algorithm

---

1 Paired x_h with its printed template x_p by compiling its label y to get the training set $((x_h, x_p), y) \in (X, Y)$;

2 Initialize the encoder-decoder model and the discriminator randomly with parameters ϕ_E and ϕ_D;

3 **repeat**

4   //Update the encoder-decoder model

5   Sample minibatch of m pairs of samples $\{(x_h, x_p)^{(1)}, \ldots, (x_h, x_p)^{(m)}\}$ from the training set;

6   Update the encode-decoder model by:
$$\phi_E \leftarrow \phi_E - Adam(\frac{\partial(\mathcal{L}_{C_h} + \mathcal{L}_{C_p} + \lambda\mathcal{L}_D)}{\partial\phi_E}, \eta_E);$$

7   //Update the discriminator for k steps

8   **for** k *steps* **do**

9   Sample minibatch of m pairs of samples $\{(x_h, x_p)^{(1)}, \ldots, (x_h, x_p)^{(m)}\}$ from the training set;

10   Update the discriminator by: $\phi_D \leftarrow \phi_D + Adam(\frac{\partial\mathcal{L}_D}{\partial\phi_D}, \eta_D)$;

11   **end**

12 **until** L_{C_h} *converged*;

13 //Get the final model for the handwritten-image-to-markup generation

14 Parameterize the encoder-decoder model by: ϕ_E;

15 **return** *The encoder-decoder model*;

---

4   Experiments

4.1   Datasets

We validate our proposal on handwritten-image-to-LaTeX generation with the large public dataset available from the Competition on Recognition of Online Handwritten Mathematical Expressions (CROHME) [22]. CROHME 2013 dataset consists of 8,836 training samples and 671 test samples. The training set of CROHME 2014 is same as CROHME 2013, but the 986 handwritten samples of the test set are newly collected and labeled. We use the CROHME 2013 test set as the validation set to estimate our model during training process and test the final model on the CROHME 2014 test set. The number of symbol classes for both the CROHME 2013 and CROHME 2014 are 101. Each mathematical expression in the dataset is stored in InkML format, which contains the trajectory coordinates of the handwritten strokes and the LaTeX and MathML format markup ground truth. Models for handwritten-image-to-LaTeX generation are evaluated at expression level by the expression recognition rate (ExpRate), which is the index that ranks the participate systems in all the CROHME competitions. A markup generation of the input image is right if the markup for all the symbols and spatial operators are generated correctly with the right order. This expression level metric is useful to evaluate all the symbols and their structures are translated rightly.

In this study, we have not used the online trajectory information of the strokes, we just connect adjacent coordinate points in the same strokes to get the offline images of the handwritten MEs. Each printed image template of training data is simply gotten by compiling the LaTeX format label with a general LaTeX editor. Then, all the images are normalized to the height of 128 pixels. And images in each minibatch are all padded to the same width as the largest one with background pixels. We use the preprocessing to ensure that the features extracted from different images are the same size.

Table 1. Configurations of the PAL model

Input: $H(128) \times W \times D(1)$ binary image	
Encoder	
CNN Block	[3 × 3 conv-32, BN, ReLU] × 4
MD-transition Layer	MaxPooling, MD-LSTM-64
CNN Block	[3 × 3 conv-64, BN, ReLU] × 4
MD-transition Layer	MaxPooling, MD-LSTM-64, Dropout 0.2
CNN Block	[3 × 3 conv-64, BN, ReLU] × 4
MD-transition Layer	MaxPooling, MD-LSTM-128, Dropout 0.25
CNN Block	[3 × 3 conv-128, BN, ReLU] × 4
MD-transition Layer	MaxPooling, MD-LSTM-256, Dropout 0.35
Decoder	
Conv-Attention	[3 conv-256, GLU, Dropout 0.5] × 3
MLP Layer	256 units, Dropout 0.5
MLP Layer	L units, Dropout 0.5, Softmax
Discriminator	
MLP Layer	512 units, ReLU, Dropout 0.2
MLP Layer	1 unit, Sigmoid

4.2   Model Configurations

In this section, we briefly summarize the configurations of our proposed PAL model. See details in Table 1. The encoder model is adapted from the deep FCN of WAP [29], but equipped with a MD-transition layer after each CNN block. Each CNN block of the encoder contains four convolutional layers, and each convolutional layer is equipped with a batch normalization layer [17] and a rectified linear unit (ReLU) [19]. The filter size of the convolutional layers is 3 × 3 and convolution stride size is 1 × 1. When a feature map is input to the hidden convolutional layer, it is zero-padded by one pixel to keep the size fixed. In addition to the size, channels of the feature maps are also fixed within the CNN blocks. Every pooling layer in the MD-transition layer is set as max-pooling with 2 × 2 kernel and 2 × 2 stride to reduce the size of the feature map.

The decoder model consists of 3 Conv-Attention blocks and a subsequent multi-layer perception (MLP). CNN block in the Conv-Attention model contains a one-dimensional convolutional layer with kernel width 3 and stride size 1.

And the one-dimensional convolutional layer is equipped with a GLU nonlinear activation function introduced in the Sect. 2.2. The discriminator D is a MLP with two fully connected layers. We employ dropout for our proposal to prevent the over-fitting. L in the table denotes the total class number of the symbols in the markup ground truth. All models are implemented in Torch and trained on 4 Nvidia TITAN X GPUs.

Table 2. ExpRate (%) of different systems on CROHME 2014 test set

System	ExpRate (%)	≤1 (%)	≤2 (%)	≤3 (%)
I	37.22	44.22	47.26	50.20
II	15.01	22.31	26.57	27.69
IV	18.97	28.19	32.35	33.37
V	18.97	26.37	30.83	32.96
VI	25.66	33.16	35.90	37.32
VII	26.06	33.87	38.54	39.96
WYGIWYS*	28.70	-	-	-
End-to-end	35.19(18.97)	-	-	-
WAP*	44.42	58.40	62.20	63.10
PAL	39.66	56.80	65.11	70.49
PAL*	**47.06**	**63.49**	**72.31**	**78.60**

4.3 Validation on CROHME

We compare our proposal with the submitted systems from CROHME 2014 and some attention-based models presented recently for handwritten-image-to-LaTeX generation. The results of these systems are listed in Table 2. Systems I to VII are the participants in CROHME 2014 and the next three systems from WYGIWYS* to WAP* are attention-based models presented recently. To make fair comparison, system III are erased from Table 2 because it has used unofficial extra training data and the attention models listed in Table 2 are all trained with offline images. The ExpRate $\leq 1(\%), \leq 2(\%), \leq 3(\%)$ denote the accuracy for markup generation with one to three symbol-level error and showing the room for the models to be further improved.

Our proposed PAL model outperforms system I, which wins the first place on CROHME 2014 and named *seehat*, with more than 2% ExpRate. More importantly, it is interesting to observe that the one to three symbol-level error of our proposal has been significantly reduced due to the grasp of semantic-invariant features for each symbol. The sign * in Table 2 denotes utilizing an ensemble of 5 differently initialized models to improve the performance [29]. WYGIWYS is the first attention-based model proposed for image-to-markup generation [11]. WYGIWYS with ensemble methods finally achieves an ExpRate of 28.70%. The End-to-end indicates encoder-decoder model in [20], which has a similar

architecture to WYGIWYS. It achieves a ExpRate of 18.97%, and 35.19% then by distorting the training images and bringing the number of training images to 6 times. WAP* here indicates the state-of-the-art model WAP [29] trained with ensemble methods and not uses the online trajectory information like other attention-based model here. We use the same ensemble method to get PAL*, and the result shows our proposed PAL model outperforms WAP under the same conditions. While the ensemble method can effectively improve performance, it requires much more memory space to run as fast as a single model through parallel computing.
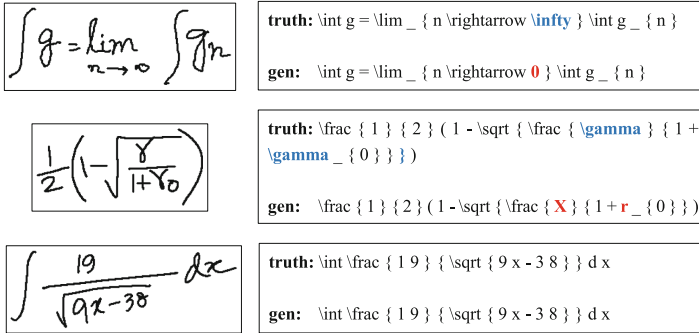

truth: \int g = \lim _ { n \rightarrow **\infty** } \int g _ { n }

gen: \int g = \lim _ { n \rightarrow **0** } \int g _ { n }

truth: \frac { 1 } { 2 } (1 - \sqrt { \frac { **\gamma** } { 1 + **\gamma** _ { 0 } } })

gen: \frac { 1 } { 2 } (1 - \sqrt { \frac { **X** } { 1 + **r** _ { 0 } } })

truth: \int \frac { 1 9 } { \sqrt { 9 x - 3 8 } } d x

gen: \int \frac { 1 9 } { \sqrt { 9 x - 3 8 } } d x

Fig. 3. Examples of the handwritten images and generated markup of our proposal

In order to make the results more intuitive, in Fig. 3 we show some handwritten MEs of the CROHME 2014 test set as well as their markup generated by our proposal. The red symbols of the *gen* indicate the incorrect markup generations, and the blue symbols of the *truth* indicate the corresponding right markup in the ground truth or the markup our proposal failed to generate. The results show our proposal are effective in dealing with the complex 2D structures and the symbols with various writing styles. It is worth noting that some symbols are too similar or written too scribbled, even humans could be confused.

4.4   Comparison of Different λ

In this section, we further analyze how the hype-parameter λ in Eq. (11) affects the performance of our proposed PAL model. By balancing the influence of the loss for markup generation and features discrimination, λ controls the trade-off between discriminative features and semantic-invariant features the encoder-decoder model learned. When λ is small, discriminative features comprise the majority loss of the encoder-decoder model and dominate the gradient backward to it. With the increasing of λ, the encoder-decoder model masters more semantic-invariant features of same symbols in the printed templates and the handwritten images. However, when λ going too large, the model will focus too much on semantic-invariant features and even try to generate same feature

sequences for both printed and handwritten images to confuse the discriminator. This will lead to less discriminative features for different categories and cause the decreasing of markup generation accuracy. For an extreme case, the model may only pays attention to regions of the background and even generates $G(x)$ that equals to a constant at each step to fool the discriminator D. Therefore, an appropriate λ plays an important role in the PAL model. We explore different λ for the model while keeping the other configurations of the model fixed the and then evaluate these models with different λ on the CROHME dataset. The results is shown in Fig. 4.
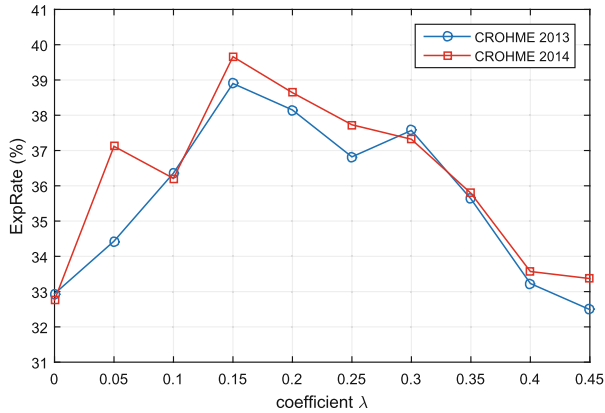
Fig. 4. Comparison of different λ on CROHME dataset

4.5 Analysis of Print Templates

It is worth noting that print templates for training are also crucial to the PAL model. Firstly, just like we cannot use printed English books to teach humans to recognize handwritten Chinese characters, the templates need to have related semantic information with the target images. Thus, the attention-based encoder-decoder model can learn semantic-invariant features for each specific symbol in the paired images. Secondly, the distribution of standard templates needs to be easier to learn. In this way, simple templates can guide the model in dealing with complex samples through paired adversarial learning.

We first validate the ability of Conv-Attention based encoder-decoder model without paired adversarial learning to generate markup for the print images. Specifically, we compile the LaTeX format markup ground truth of CROHME 2013 test set to get the printed CROHME 2013 test set (CROHME 2013 P) and train the model only on the standard printed templates, see Conv-Attention P in Table 3. Then we get the same model but trained on only the handwritten images (Conv-Attention H). Surprisingly, the accuracy of printed-image-to-markup generation is more than double of the handwritten-image-to-markup generation.

Table 3. Analysis of the influence of printed templates

ExpRate (%)	CROHME 2013	CHROHME 2014	CROHME 2013 P
Conv-Attention H	31.30	27.18	-
Conv-Attention P	0.15	-	76.15
Conv-Attention P&H	32.94	33.87	-
PAL	38.90	39.66	57.82
PAL GD	41.73	39.05	-

However, the model Conv-Attention P appears to have been over fitted in the printed images when tested on the handwritten CROHME 2013 test set.

After that, we mix these two samples to conduct experiments (Conv-Attention P&H). The experimental results show that the distributions of these two kinds of samples is relatively close, and the adding of the printed templates is helpful to the markup generation for the handwritten images. The model increased about 1% ExpRate compared with Conv-Attention H when validated on CROHME 2013 test set and the generalization is significantly enhanced when test on CROHME 2014 test set. When we train the model with paired adversarial learning (PAL) and set λ as 0.15, we find that this increase becomes even more apparent, whether it is validation or test. We also tested the PAL model on CROHME 2013 P, the result shows that the model does lose some knowledge about the distribution of the print images compared with Conv-Attention P.

In addition, we have made some global distortions for the printed templates to further explore the influence of the print templates' distribution. We rote the standard printed sample with 4 angles randomly choose from $-2°$ to $2°$ with a interval of $0.5°$ but excluding $0°$ (the minus sign here represents counterclockwise). Then we add them to the standard printed templates without distortion and re-pair each of these printed templates with the handwritten one that owned the same label. The new $8,836 * 5$ image pairs are used to train the model called PAL GD. Interestingly, we find that the accuracy of the validation has been further improved, but the accuracy of the test has slightly decreased. It is believed that if the distortions are done more elaborately, the test accuracy will also be improved, but this contradicts our original intention of training the attention-based model through easy-to-get templates with paired adversarial learning. Therefore, we haven't conducted further experiments on the distortion.

5   Conclusion

In this paper, we introduce a novel paired adversarial learning to guide the attention-based model to learn the semantic-invariant features as human when focusing attention on specific objects. Our proposal incorporates the prior knowledge of simple templates and improves the performance of an attention-based model on more complex tasks. The proposal performs much better than other

systems under the same training conditions on CROHME 2014. We also extend a fully convolutional attention from machine translation to speed-up the decoding of the image-to-markup generation.

In future work, we plan to explore the language model based attention to make the neural network more like human when generating the markup for the input images. We will also apply the paired adversarial learning in more fields such as text string recognition in optical character recognition to improve the models performance.

References

1. Álvaro, F., Sánchez, J.A., Benedí, J.M.: An integrated grammar-based approach for mathematical expression recognition. Pattern Recogn. **51**, 135–147 (2016)
2. Anderson, R.H.: Syntax-directed recognition of hand-printed two-dimensional mathematics. In: Symposium on Interactive Systems for Experimental Applied Mathematics: Proceedings of the Association for Computing Machinery Inc. Symposium, pp. 436–459. ACM (1967)
3. Awal, A.M., Mouchère, H., Viard-Gaudin, C.: A global learning approach for an online handwritten mathematical expression recognition system. Pattern Recogn. Lett. **35**, 68–77 (2014)
4. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
5. Bousmalis, K., Silberman, N., Dohan, D., Erhan, D., Krishnan, D.: Unsupervised pixel-level domain adaptation with generative adversarial networks. In: The IEEE Conference on Computer Vision and Pattern Recognition, pp. 95–104 (2017)
6. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: InfoGAN: interpretable representation learning by information maximizing generative adversarial nets. In: Advances in Neural Information Processing Systems, pp. 2172–2180 (2016)
7. Cho, K., Courville, A., Bengio, Y.: Describing multimedia content using attention-based encoder-decoder networks. IEEE Trans. Multimedia **17**(11), 1875–1886 (2015)
8. Chorowski, J.K., Bahdanau, D., Serdyuk, D., Cho, K., Bengio, Y.: Attention-based models for speech recognition. In: Advances in Neural Information Processing Systems, pp. 577–585 (2015)
9. Dauphin, Y.N., Fan, A., Auli, M., Grangier, D.: Language modeling with gated convolutional networks. arXiv preprint arXiv:1612.08083 (2016)
10. Deng, Y., Kanervisto, A., Ling, J., Rush, A.M.: Image-to-markup generation with coarse-to-fine attention. In: International Conference on Machine Learning, pp. 980–989 (2017)
11. Deng, Y., Kanervisto, A., Rush, A.M.: What you get is what you see: a visual markup decompiler. arXiv preprint arXiv:1609.04938 (2016)

12. Denton, E.L., Chintala, S., Fergus, R., et al.: Deep generative image models using a Laplacian pyramid of adversarial networks. In: Advances in Neural Information Processing Systems, pp. 1486–1494 (2015)
13. Gehring, J., Auli, M., Grangier, D., Yarats, D., Dauphin, Y.N.: Convolutional sequence to sequence learning. arXiv preprint arXiv:1705.03122 (2017)
14. Goodfellow, I., et al.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, pp. 2672–2680 (2014)
15. Graves, A., Schmidhuber, J.: Offline handwriting recognition with multidimensional recurrent neural networks. In: Advances in Neural Information Processing Systems, pp. 545–552 (2009)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: The IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
17. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)
18. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. arXiv preprint (2017)
19. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
20. Le, A.D., Nakagawa, M.: Training an end-to-end system for handwritten mathematical expression recognition by generated patterns. In: 14th International Conference on Document Analysis and Recognition, vol. 1, pp. 1056–1061. IEEE (2017)
21. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
22. Mouchere, H., Viard-Gaudin, C., Zanibbi, R., Garain, U.: ICFHR 2014 competition on recognition of on-line handwritten mathematical expressions (CROHME 2014). In: 14th International Conference on Frontiers in Handwriting Recognition, pp. 791–796. IEEE (2014)
23. Mouchere, H., Zanibbi, R., Garain, U., Viard-Gaudin, C.: Advancing the state of the art for handwritten math recognition: the CROHME competitions, 2011–2014. Int. J. Doc. Anal. Recogn. **19**(2), 173–189 (2016)
24. Qureshi, A.H., Nakamura, Y., Yoshikawa, Y., Ishiguro, H.: Show, attend and interact: perceivable human-robot social interaction through neural attention q-network. In: International Conference on Robotics and Automation, pp. 1639–1645. IEEE (2017)
25. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
26. Sayre, K.M.: Machine recognition of handwritten words: a project report. Pattern Recogn. **5**(3), 213–228 (1973)
27. Sukhbaatar, S., Weston, J., Fergus, R., et al.: End-to-end memory networks. In: Advances in Neural Information Processing Systems, pp. 2440–2448 (2015)
28. Zhang, J., Du, J., Dai, L.: A gru-based encoder-decoder approach with attention for online handwritten mathematical expression recognition. arXiv preprint arXiv:1712.03991 (2017)
29. Zhang, J., et al.: Watch, attend and parse: an end-to-end neural network based approach to handwritten mathematical expression recognition. Pattern Recogn. **71**, 196–206 (2017)

30. Zhou, X.D., Wang, D.H., Tian, F., Liu, C.L., Nakagawa, M.: Handwritten Chinese/Japanese text recognition using semi-Markov conditional random fields. IEEE Trans. Pattern Anal. Mach. Intell. **35**(10), 2413–2426 (2013)
31. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. arXiv preprint arXiv:1703.10593 (2017)