AI Drawing Chinese Character

Heri Zhao@Yahoo   Jiayu Ye@Google   Ke Xu@Stanford EE

Introduction

Chinese character are widely adopted in Asian. How AI will apply to this Intangible Cultural Heritage will be a juicy topic. Inspired by the DRAW[1] and it's application on chinese characters[2], we are provisioning a joint model to take the calligraphy style into account as artistic style was applied on image[3]. At first we can deal with a small but curial problem, how we can predict the correct strokes sequence for a new character after we input a set of known character first.

Evaluation Metrics

Since the correct sequence for all characters are known (from dictionary, etc), the evaluation metric would be to use a image reconstruction quality. A straightforward loss function will be

$$\alpha \sum_{each\ stroke} distance(predicted\ location,\ ground\ truth) + \beta \sum_{each\ stroke} (predicted\ seq\#\ != \ ground\ truth)$$

Another option to evaluate will be the classification mean average precision for generated character. This method is from a competition dataset as used in paper[2] and icdar 2013[4].

For the exact dataset that baseline solution has dependent on, please refer to the Baseline section.

Input & Output examples:

Below is the example for **oracle**:
Input: 永 (.svg)
Outpt: calligraphy drawing animation, example.

For **experimental** stage, several domain specific information are optionally incorporated.
For example:

Stroke image Sequence: 
Location sequence: Absolute location of each stroke.

[1] DRAW: A Recurrent Neural Network For Image Generation, https://arxiv.org/pdf/1502.04623.pdf
[2] Drawing and Recognizing Chinese Characters with Recurrent Neural Network, https://arxiv.org/abs/1606.06539
[3] A Neural Algorithm of Artistic Style, https://arxiv.org/abs/1508.06576
[4] http://www.nlpr.ia.ac.cn/events/CHRcompetition2013/competition/Home.html

Chinese character written rules: left to right, top to bottom, e.t.c

Baseline explanation

Dataset

For each given character, the data contains a list of partial stokes in order (**Online Dataset**). For example, the character "十" (horizontal line and vertical line) has two strokes:

["#1PR:[(x1, y1)...]","#3PR:[(x2, y2)...]"]

Each (partial) stroke data string starting by a “#” and followed by:

1. A number: the direction of the stroke
   - 1: Left to Right,
   - 2: Upper Left to Lower Right
   - 3: Top to Bottom
   - etc.
2. Ignore PR for now.
3. ~~List of polygon points in the form (x, y) separated by “;”. (Not used yet)~~

In the baseline model, we only consider each stroke's direction. The above example “十” corresponds to the input as a list of stroke directions, [[1], [3]]

Algorithm

The algorithm is a strokes-order search problem, which will be based on the bigram cost given by the training set. We constructed a strokeReorderProblem:

- State: (pre_stroke, list of strokes that has not yet been selected)
- Start state: the first stroke ground truth
- End state: the strokes list is empty
- succAndCost: Given a state, return (action, newState, cost) based on the bigram cost

Result

Apply uniform cost search on this strokeReorderProblem, the result:
Number of characters in training dataset: 185

#Stroke in Char	#Correct	#Total	Bigram Accuracy	Random Accuracy
3	19	27	70.37%	1/(3-1)! = 50%
4	16	54	29.63%	1/(4-1)! = 16.67%
5	9	69	13.04%	1/(5-1)! = 4.16%
6	2	111	1.8%	1/(6-1)! = 0.83%

We did not include the test characters that only have 1 or 2 strokes, because given the true start state as prior knowledge, there is no point to predict for 1 or 2 strokes. We also did not include the test characters that contains more than 6 strokes, because the accuracy is so low. As we can see, the bigram-algorithm is much more better than the theoretical accuracy of the random algorithm.

Oracle

A good oracle model is human level chinese characters drawing. Based on massive training and education, a person can learn the basic rules of writing characters. For example, chinese characters generally are written from upper-left to lower-right. More specifically, a person is able to know how to write chinese character components instead of individual strokes, and then apply the general rules to combine all the components together by order.

Challenges

1. How to distinguish different types of strokes?
   The directions dataset are just describing high-level features like left to right, which is likely to be two different stroke types. To address this challenge, using a sliding window based recognition algorithm to match each stroke with it's specific type.
2. How to use the location of each strokes to help predict?
   In the baseline experiment, we ignored the location information. Location information can be a heuristic for this search model.
3. How to solve all the problems together.?
   A recurrent neural network will be an state-of-art approach to tackle the problem incorporating sequential information, as proved in [2].

Related Work: Addressed in the introduction