Interpretable Distance Metric Learning for Handwritten Chinese Character Recognition

Boxiang Dong*, Aparna S. Varde*, Danilo Stevanovic* Jiayin Wang*, Liang Zhao†
*{dongb, vardea, stevanovicd1, jiayin.wang}@montclair.edu
Montclair State University, Montclair, New Jersey 07043
† liangzhao@dlut.edu.cn
Dalian University of Technology, Dalian, China 116024

***Abstract*—Handwriting recognition is of crucial importance to both Human Computer Interaction (HCI) and paperwork digitization. In the general field of Optical Character Recognition (OCR), handwritten Chinese character recognition faces tremendous challenges due to the enormously large character sets and the amazing diversity of writing styles. Learning an appropriate distance metric to measure the difference between data inputs is the foundation of accurate handwritten character recognition. Existing distance metric learning approaches either produce unacceptable error rates, or provide little interpretability in the results. In this paper, we propose an interpretable distance metric learning approach for handwritten Chinese character recognition. The learned metric is a linear combination of intelligible base metrics, and thus provides meaningful insights to ordinary users. Our experimental results on a benchmark dataset demonstrate the superior efficiency, accuracy and interpretability of our proposed approach.**

***Index Terms*—Big Data, Distance Components, HCI, Machine Learning, OCR, Pictorial Characters, Text Recognition**

I. Introduction

Handwriting recognition is gaining increasing importance given the prevalence of mobile devices and tablets. A majority of people still prefer handwritten input over keyboard entry, especially when taking notes in a classroom or meeting, and annotating a digital document. The need for an accurate and reliable handwriting recognition solution is even stronger among Chinese users, given the fact that it is extremely time-consuming to enter a Chinese character via keyboard [1]. In particular, users have to type in the pronunciation (i.e., Pinyin) of the desired Chinese character first, and then choose the target from a list of candidates. To make it even more difficult, the pronunciation of a single Chinese character usually consists of at least 4 English characters. On the other hand, handwriting recognition is the fundamental technology of Optical Character Recognition, which is widely applied in handwritten check clearance and judicial paperwork digitization.

Pairwise distance metric, a function that measures the dissimilarity between a pair of data inputs, plays crucial role in handwritten character recognition. Ideally, it helps to identify the handwriting inputs that correspond to the same character. It is obvious that the performance of a handwritten Chinese character recognition (HCCR) system heavily depends on the quality of the underlying distance metric. Distance metric learning (DML) aims at automatically learning an appropriate distance (or similarity) measure from labeled samples [2], [3]. Recent results [4] reveal that even a simple linear transformation of the input features can significantly improve the classification accuracy. Therefore, DML provides a natural solution to determine the distance metric for handwritten Chinese character recognition.

Surprisingly, [5] is the only work on distance metric learning for handwritten Chinese character recognition (based on our literature search). However, the application of the distance metric learning there is only limited to text line segmentation. Handwritten Chinese character recognition and distance metric learning do face their own challenges respectively. In terms of handwritten Chinese character recognition, the challenges arise from the enormous character set and the diversity of writing styles. Unlike alphabet-based writing, which typically comprises the order of 100 symbols, there are 27,533 entries in Chinese National Standard GB18030-2005. Moreover, the divergence of writing styles among different writers and in different geographic areas aggravates the confusion between different characters [6]. These difficulties lead to unsatisfactory performance in handwriting recognition. For example, [7] can only provide 39.37% recognition accuracy on a dataset with 186,444 characters. Later works [8], [9] improve the accuracy up to 78.44% and 73.87% respectively. Regarding distance metric learning, most works focus on learning a Mahalanobis distance metric. Even though it is equivalent to computing the Euclidean distance after a linear projection of the data, it provides limited interpretability. In particular, the learned transformation matrix cannot explain the relative importance of the features in the distance metric. Lacking interpretability prohibits further analysis in the case of misclassification and undermines user confidence.

In this paper, our objective is to put forth an interpretable distance metric learning approach for accurate offline handwritten Chinese character recognition. By offline, we mean that the focus is on recognizing characters already written on paper earlier. The input is in the form of a scanned image of the paper document. Compared with online recognition, offline character recognition is more challenging in that it does not have the trace of the writer's pen as well as the order of writing in the input. It has been shown [10] that such pen dynamics information can help to obtain better recognition accuracy than static scanned images alone.

To provide interpretability in the learned distance metric,

978-1-7281-4117-6/19/$31.00 ©2019 IEEE
DOI 10.1109/DataCom.2019.00025

we firstly define a set of base distance metrics, which we call *components* in the rest of the paper. These components quantify the dissimilarity of two handwritten characters in a simple manner. They can be provided by domain experts in fields such as Linguistics, or can be proposed based on a preliminary analysis of the data.

The components used in our experiments include the difference of the length of the longest horizontal stroke, the longest vertical stroke, etc. Given these components, we propose an ensemble learning strategy to linearly combine these basic metrics into a strong distance metric, so as to guarantee accurate handwritten character recognition.

To the best of our knowledge, ours is among the first works to learn an interpretable distance metric for handwritten Chinese character recognition. In particular, this work makes the following contributions.

- We design a new algorithm named *MetChar* that optimizes the weight assignment for a given set of basic components so as to obtain a distance metric, which is later used by the clustering algorithm to classify the input handwritten characters. *MetChar* follows a style analogous to the gradient descend optimizer [11], but it copes with the fact that the error rate is not differentiable. A good property of *MetChar* is that it is compatible with a wide range of clustering algorithms.
- We propose an approach, namely *HybridSelection*, that chooses the combination of basic components from a large candidate pool, and feeds them to *MetChar* for optimization. The *HybridSelection* algorithm trims off the components that do not meet the quality requirement, and fully takes advantage of the remaining ones. It reaches a balance between efficiency and accuracy.
- We run a set of experiments on a benchmark dataset. The results demonstrate the superiority of *HybridSelection*. It produces the highest recognition accuracy in an affordable time. Besides, the learned distance metric is interpretable for ordinary users.

The rest of the paper is organized as follows. Section II reviews the preliminaries. Section III presents our distance metric learning approaches. Section IV discusses the experimental results. Section V introduces the related work. Finally, Section VI concludes the paper.

II. Preliminaries

A. Individual Distance Components

In computer vision and data management, it is well known that a small fraction of the regions in a figure can hold critical information for object recognition and classification [12]–[16]. In addition, certain statistical studies on the figure depict the contextual behavior in the image in a succinct manner [17]. In the field of handwritten Chinese character recognition, we have similar observations. Take the handwritten characters in Fig. 1 as an example. The length and position of the longest horizontal and vertical strokes in a handwritten character can serve as effective and convenient features for the purpose of recognition. Next, the individual distance components are defined by applying simple operators on these intelligible features, such as Euclidean distance and Manhattan distance. As opposed to sophisticated feature engineering techniques of

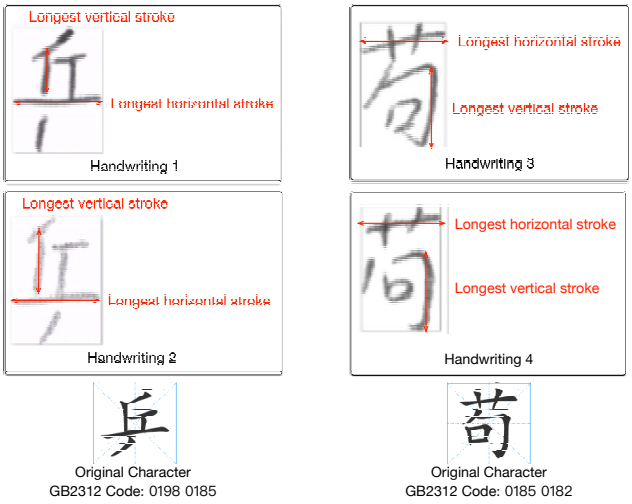

Fig. 1. An example of individual distance components in handwritten Chinese character recognition

crossings and celled projections [18], [19], the components used in this paper are basic ones that incorporate fundamental human intelligence, and are more easily interpretable.

B. Distance Metric Learning

In this paper, we aim at proposing an interpretable distance metric learning approach for handwritten Chinese character recognition. To provide interpretability, our objective is to learn a distance metric that is a linear combination of the intelligible individual components. In particular, given a set of individual components (metrics) $\boldsymbol{d} = \{d_1, \ldots, d_p\}$, where $d_i(\boldsymbol{x}, \boldsymbol{x}')$ is per the discussion in Section II-A, the target distance metric is

$$D(\boldsymbol{x}, \boldsymbol{x}') = \sum_{i=1}^{p} w_i d_i(\boldsymbol{x}, \boldsymbol{x}'), \tag{1}$$

where $w_i \geq 0$ for every $1 \leq i \leq p$, $\boldsymbol{x}$ and $\boldsymbol{x}'$ are a pair of handwriting characters. Due to the linear combination in Equation (1), the weight associated with each basic component signifies its importance in handwritten character recognition, so as to provide interpretability in the decision/classification result. It can be clearly seen that our objective is indeed a bagging algorithm to combine multiple basic but intelligible individual components, so as to formulate an accurate and interpretable distance metric.

III. Distance Metric Learning Approach

In our approach, we first propose an algorithm named *MetChar* that optimizes the weight assignment for a fixed set of components. Next, we propose an algorithm named *HybridSelection* to select components as the input to *MetChar*.

A. *The* MetChar *Algorithm*

MetChar recognizes/classifies the input handwritten characters based on clustering. It is worth noting that *MetChar* is compatible with any clustering algorithm. Given a set of individual components $d_1, \ldots, d_q \subseteq \boldsymbol{d}$, where $\boldsymbol{d}$ is the complete set of available components, *MetChar* produces the component weights $w_1, \ldots, w_q$ so that the clustering algorithm based on the distance metric $D(\boldsymbol{x}, \boldsymbol{x}') = \sum_{i=1}^{q} w_i d_i(\boldsymbol{x}, \boldsymbol{x}')$ assigns the handwriting samples of the same character into the same cluster, and those of different characters into different clusters. Before explaining the optimization procedure, we first present a few definitions that are needed later. Given a pair of handwriting samples $\boldsymbol{x}$ and $\boldsymbol{x}'$, let their corresponding characters be y and y' respectively. Let c and c' denote the clusters to which they are assigned. We then have the following definitions based on the relationship between the characters and the clusters.

- $(\boldsymbol{x}, \boldsymbol{x}')$ is a true positive (TP) if $y = y'$ and $c = c'$.
- $(\boldsymbol{x}, \boldsymbol{x}')$ is a true negative (TN) if $y \neq y'$ and $c \neq c'$.
- $(\boldsymbol{x}, \boldsymbol{x}')$ is a false positive (FP) if $y \neq y'$ but $c = c'$.
- $(\boldsymbol{x}, \boldsymbol{x}')$ is a false negative (FN) if $y = y'$ but $c \neq c'$.

It is obvious that TPs and TNs are the pairs that are correctly recognized. Hence we define the accuracy as $acc = \frac{TP+TN}{TP+TN+FP+FN}$.

In order to improve the recognition accuracy, the intuition is to increase the pairwise distance for the FPs, while reducing the distance for the FNs. At the t-th round of the learning stage, let $D^{(t)}$ be the learned distance metric, and $w_i^{(t)}$ denote the weight associated with the individual component d_i. For any d_i, we can calculate two values: $\alpha_i^{(t)}$ which denotes the total distance on d_i for the FPs, and $\beta_i^{(t)}$ which denotes the total distance on d_i for the FNs. In particular, we have

$$\alpha_i^{(t)} = \sum_{(\boldsymbol{x}, \boldsymbol{x}') \in FP} w_i^{(t)} d_i(\boldsymbol{x}, \boldsymbol{x}'), \tag{2}$$

and

$$\beta_i^{(t)} = \sum_{(\boldsymbol{x}, \boldsymbol{x}') \in FN} w_i^{(t)} d_i(\boldsymbol{x}, \boldsymbol{x}'). \tag{3}$$

In the next round, i.e., the $(t+1)$-th round, we update the weight associated with d_i by:

$$w_i^{(t+1)} = max(0, w_i^{(t)} + \epsilon(\alpha_i^{(t)} - \beta_i^{(t)})), \tag{4}$$

where ϵ is a given learning rate. We repeatedly update the weights $w_1, \ldots, w_q$ for a certain number of iterations. Algorithm 1 displays the pseudocode for *MetChar*.

MetChar follows a style similar to the classical gradient descent algorithm [11], which is widely adapted in deep learning. In particular, they both start with a random initialization of the weights, and then gradually optimize the weights throughout the iterations. However, since the underlying recognition algorithm is clustering, the loss value or error rate is not differentiable over the weights. In other words, it is impossible to calculate the derivatives of the loss over the weights. Therefore, unlike the original gradient descent algorithm, *MetChar* refines the weights by taking the FPs and FNs into consideration (Equation (2 - 4)).

Algorithm 1 *MetChar*

Input: A set of individual components $d_1, \ldots, d_q$, the training set $\{(\boldsymbol{x}, y)\}$, the learning rate ϵ, the number of iterations T, the number of unique characters k
Output: The distance metric $D = \sum_{i=1}^{q} w_i d_i(\boldsymbol{x}, \boldsymbol{x}')$

```
Randomly assign initial weights w_1^(1), ..., w_q^(1)
for t = 1 to T do
    Let D^(t) = Σ_{i=1}^{q} w_i^(t) d_i(x, x')
    Apply the clustering algorithm with D^(t) to get k clusters
    Calculate the accuracy acc^(t)
    for i = 1 to q do
        Update w_i^(t+1) according to Equation (4)
    end for
end for
Let t* be the round that produces the highest acc^(t*)
return D^(t*)
```

B. *Component Selection Algorithms*

The *MetChar* algorithm takes a set of individual components as the input, and by default utilizes all of them to learn a distance metric for handwritten Chinese character recognition. However, in practice, it is not necessary to exploit all the components, especially if a large candidate pool is available. Introducing redundant components demands longer optimization time of *MetChar*, while bringing minimal accuracy benefits. Moreover, since the number of parameters to be optimized is linear to the number of components, redundant components could make the learned distance metric overfit the training data, and lead to a suboptimal solution [20]. Therefore, it is imperative to have an approach for selecting a subset of components from the candidate pool which *MetChar* can employ to deliver satisfactory recognition accuracy.

A naive solution is to enumerate all possible combinations of components and feed them to *MetChar*. However, this incurs significant computational overhead. The search complexity is exponential to the number of components, i.e. $O(2^p)$. This is overwhelming when p is a large number. To reach a balance between efficiency and accuracy, we propose *HybridSelection* that firstly eliminates the least promising candidate components, based on a given error threshold, and then examines all the combinations of the remaining components.

Algorithm 2 shows the pseudocode. From Line 1 - 5, *HybridSelection* evaluates the quality of each individual component. If d_i does not meet the quality requirement, i.e., $acc_i < \theta$ (error threshold), it is eliminated from the component pool. From Line 6 to 10, *HybridSelection* tries every combination of the remaining components, and finds the one with the highest accuracy. Let s denote the number of remaining components after the first loop, the complexity is $O(p + 2^s)$, where $s < p$. Therefore, the complexity is lower than that of *ExhaustiveSelection*. Meanwhile, we can adjust the error

Algorithm 2 *HybridSelection*

Input: A complete set of individual components $\boldsymbol{d} = \{d_1, \ldots, d_p\}$, an error threshold θ
Output: The distance metric D

```
for j = 1 to p do
    Run MetChar (Algorithm 1) with d_i only
    Evaluate the accuracy acc_i
    Prune d_i from d if acc_i < θ
end for
for j = 2 to p do
    for each combination d' of j components in d do
        Call MetChar (Algorithm 1) on d'
    end for
end for
return the distance metric with the highest accuracy
```

threshold θ to alternate the balance between efficiency and accuracy. A larger θ results in a smaller s, and thus better efficiency but potentially lower accuracy. Our experimental results in Section IV demonstrate that *HybridSelection* delivers satisfactory accuracy. This is because the search space discarded by *HybridSelection* only includes the least promising combination of components, and thus induces little impact to the recognition accuracy. Given the same time constraints, the accuracy of *HybridSelection* can even be higher than that of the exhaustive search algorithm.

TABLE I
DESCRIPTION OF THE CASIA-HWDB1.1 DATASET

Dataset	# of Writers	# of Classes	# of Sample Images
CASIA-HWDB1.1	300	3,755	1,121,749

IV. EXPERIMENTS

A. Setup

We implement the distance metric learning approaches in Java. We use k-means clustering in the *MetChar* algorithm. The source code is publicly available[1]. We execute all the experiments on a MacBook Pro with 3.1 GHz Intel i5 CPU and 16 GB RAM, running Mac OS X.

B. Dataset

We run experiments on the CASIA Offline Chinese Handwriting Database V1.1[2], namely CASIA-HWDB1.1. This is built by the National Laboratory of Pattern Recognition, Institute of Automation of Chinese Academy of Sciences. The dataset is produced by 300 writers using Anoto pen on papers for obtaining offline images (in resolution of 300DPT). The collected images are segmented and annotated at the character level. The dataset includes 1,121,749 writing samples of 3,755 unique GB2312-80 level-1 Chinese characters. Table I presents the details of the CASIA-HWDB dataset.

[1]https://github.com/bxdong7/DML4HCCR
[2]http://www.nlpr.ia.ac.cn/databases/handwriting/Home.html

Every image in this dataset has its background labeled as 255 and its foreground pixels in 255 gray levels (0-254). In our experiments, we randomly choose 10 unique characters. The training and testing datasets include 30 non-overlapping sample images for each character.

C. Distance Components

We preprocess each image by simply changing the foreground pixels to 1 and the background pixels to 0 so as to obtain the binary image. From each image, we extract the following features (see Fig. 1).

- hbv: the horizontal bit vector, which denotes the number of 1s in each horizontal line;
- hfv: the horizontal first foreground bit vector, which stores the location of the first 1 in each horizontal line;
- hlv: the horizontal last foreground bit vector, which stores the location of the last 1 in each horizontal line;
- vfv: the vertical first foreground bit vector, which stores the location of the first 1 in each vertical line;
- vlv: the vertical last foreground bit vector, which stores the location of the last 1 in each vertical line;
- dfv: the diagonal first foreground bit vector, which stores the location of the first 1 in each diagonal line; and
- dlv: the diagonal last foreground bit vector, which stores the location of the last 1 in each diagonal line.

Based on these features, we have the following basic distance metric components.

- hbv_md: the Manhattan distance between the hbvs of a pair of sample images;
- hfv_md: the Manhattan distance between the hfvs of a pair of sample images;
- vfv_md: the Manhattan distance between the vfvs of a pair of sample images;
- vfv_ed: the Euclidean distance between the vfvs of a pair of sample images;
- dfv_md: the Manhattan distance between the dfvs of a pair of sample images;
- dfv_ed: the Euclidean distance between the dfvs of a pair of sample images;
- hlv_md: the Manhattan distance between the hlvs of a pair of sample images;
- vlv_md: the Manhattan distance between the vlvs of a pair of sample images;
- vlv_ed: the Euclidean distance between the vlvs of a pair of sample images; and
- dlv_ed: the Euclidean distance between the dlvs of a pair of sample images.

D. Baselines

In the experiment, we compare our component selection algorithm named *HybridSelection* with the following baselines:

- **ExhaustiveSelection** This method enumerates all possible combinations of individual distance components.
- **GreedySelection** It firstly evaluates the recognition accuracy by using single component, and produces a canonical

TABLE II
PERFORMANCE OF *HybridSelection* WITH AT LEAST THREE COMPONENTS

Components	Weights	Time (s)	Accuracy
[vlv_md, dlv_ed, hbv_md, dfv_md]	[7.766, 5.803, 2.788, 3.197E-12]	24.805	0.6650
[vlv_md, dlv_ed, hbv_md, dfv_ed]	[6.152, 10.16, 1.267, 7.782]	29.717	0.6802
[vlv_md, dlv_ed, hbv_md, vfv_md]	[6.515, 9.432, 0.08981, 3.035]	27.929	0.7218
[vlv_md, dlv_ed, dfv_md, dfv_ed]	[1.623, 5.327, 0.2230, 4.277]	31.191	0.7602
[vlv_md, dlv_ed, dfv_md, vfv_md]	[7.116, 16.63, 4.711E-4, 5.391]	26.015	0.6832
[vlv_md, dlv_ed, dfv_ed, vfv_md]	[1.385, 10.33, 7.089, 0.8096]	29.067	0.7993
[vlv_md, hbv_md, dfv_md, dfv_ed]	[3.265, 4.753, 0.002662, 13.27]	35.122	0.7960
[vlv_md, hbv_md, dfv_md, vfv_md]	[6.675, 4.803, 1.493E-13, 4.174]	58.505	0.3684
[vlv_md, hbv_md, dfv_ed, vfv_md]	[3.999, 4.336, 13.61, 0.001724]	35.008	0.6466
[vlv_md, dfv_md, dfv_ed, vfv_md]	[5.235, 2.305, 13.31, 0.86907]	39.629	0.7120
[dlv_ed, hbv_md, dfv_md, dfv_ed]	[5.642, 0.7848, 4.593E-5, 4.542]	34.291	0.6791
[dlv_ed, hbv_md, dfv_md, vfv_md]	[14.54, 0.2431, 8.6984E-5, 4.491]	31.894	0.6993
[dlv_ed, hbv_md, dfv_ed, vfv_md]	[5.751, 0.1945, 4.172, 0.4177]	36.026	0.6715
[dlv_ed, dfv_md, dfv_ed, vfv_md]	[11.55, 0.012576, 8.236, 0.9782]	33.655	0.6363
[hbv_md, dfv_md, dfv_ed, vfv_md]	[4.438, 0.008846, 13.23, 0.002252]	42.801	0.7562
[vlv_md, dlv_ed, hbv_md, dfv_md, dfv_ed]	[9.344, 12.51, 1.414, 0.03516, 9.257]	36.644	0.7512
[vlv_md, dlv_ed, hbv_md, dfv_md, vfv_md]	[3.633, 20.90, 0.6605, 0.001985, 6.226]	30.902	0.7191
[vlv_md, dlv_ed, hbv_md, dfv_ed, vfv_md]	[4.723, 8.905, 0.4436, 5.994, 0.5144]	34.465	0.7279
[vlv_md, dlv_ed, dfv_md, dfv_ed, vfv_md]	[1.460, 12.54, 0.030117, 8.365, 0.9173]	34.917	0.7089
[vlv_md, hbv_md, dfv_md, dfv_ed, vfv_md]	[5.694, 4.262, 0.001655, 12.48, 9.859E-4]	42.443	**0.8075**
[dlv_ed, hbv_md, dfv_md, dfv_ed, vfv_md]	[10.70, 0.5038, 0.003774, 7.932, 0.5845]	42.591	0.7729
[vlv_md, dlv_ed, hbv_md, dfv_md, dfv_ed, vfv_md]	[1.252, 0.5961, 19.19, 11.281, 4.382, 2.560]	41.859	0.7227

ordering of the individual components based on the accuracy. When generating a combination of $j > 2$ components, it only considers the union of the j best individual components.

E. Performance of Evaluation

Table II displays the performance of *HybridSelection* with at least three components. In our experiments, we set $\theta = 0.55$ based on grid search. This value of θ implies that those components producing a recognition accuracy lower than 0.55 are pruned out in Line 1 - 5 of Algorithm 2. This operation results in the six remaining components for consideration in Line 6 - 10 of Algorithm 2. From Table II, we can see that *HybridSelection* completes examining a majority of the candidate combinations within 10 minutes. Moreover, the accuracy, 0.8075, i.e., 80.75% (as highlighted in the table) is acceptable for use in applications.

We also observe that the learned distance metric is interpretable as follows. Based on the associated weights, we can learn that among the 5 components whose combination provides the highest accuracy, vlv_md, hbv_md and dfv_ed play the most important role. **These components yield an accuracy of 80.7$\%$, i.e. $\sim$ 81$\%$** as observed from the experiments. It implies that the *horizontal bit vector, vertical last foreground bit vector, and diagonal first foreground bit vector* are of crucial value for handwritten Chinese character recognition. These are highly meaningful insights gained from our study, with respect to metric interpretation.

In Table III, we compare the recognition accuracy delivered by the three component selection algorithms. We limit the execution time of all algorithms to be 24 hours. The result suggests that *HybridSelection* produces the highest accuracy, which is even higher than *ExhaustiveSelection*. This is because *ExhaustiveSelection* can only complete enumerating up to 4 components within the time limit, which yields 78.28% recognition accuracy at best. On the other hand, *GreedySelection* only inspects a very limited fraction of the search space. Therefore, its accuracy is the lowest. This result demonstrates the advantage of *HybridSelection* - reach a balance between time efficiency and recognition accuracy.

TABLE III
COMPARISON OF *HybridSelection* WITH THE BASELINES

Algorithm	Accuracy
ExhaustiveSelection	0.7828
GreedySelection	0.7694
HybridSelection (Our Approach)	**0.8075**

V. RELATED WORK

A. Distance Metric Learning

The need for appropriate ways to measure the distance or similarity between data points is almost ubiquitous in machine learning and data mining. This leads to the emergence of DML, which aims at automatically learning a suitable metric from data [3]. In terms of the format, there exist linear and non-linear metrics. Non-linear metrics, such as the two-histogram distance can capture non-linear variations in the data, however they give rise to non-convex formulations. Linear metrics, such as the Mahalanobis distance, are easier to optimize and thus attract much attention in metric learning.

Chang et al. [21] propose a boosting Mahalanobis distance metric (BoostMDM) method. It iteratively employs a base-learner to update a base matrix. A framework to combine base matrices is proposed in their paper, along with a base learner algorithm specific to it. The loss function describes the hypothesis margin, a lower bound of the sample margin used in methods such as SVMs (Support Vector Machines). A

problem with DML is that since the number of parameters to be determined is quadratic to the dimension of the features, it may lead to overfitting the training data [22], and provide a suboptimal solution. To resolve the problem, Qian et al. [20] propose a regularization approach that applies dropout to both the learned metrics and the training data.

The work quite closely related to ours is [22]. It aims at learning a Mahalanobis metric for clustering. The metric seeks to minimize the distance between similarly labeled inputs while maximizing the distance between differently labeled samples. However, this work does not specifically consider pictorial scripts with substantial diversity in OCR, nor does it focus on easily interpretable metrics.

B. Handwritten Chinese Character Recognition

Research on Chinese handwriting recognition of general texts has been observed only in recent years [23], and the reported accuracy is quite low. Recent works, using character classifiers and statistical language models (SLM) based on oversegmentation, obtain a character-level accuracy of 78.44% [8] and 73.97% [9], respectively. Earlier works show even lower accuracy.

Recent years witness the popularity of deep learning in fields such as natural language processing, computer vision and machine learning. Zhang et al. [24] obtain 95.88% character-level accuracy on the CASIA-HWDB dataset by using a 15-layer convolutional neural network (CNN). Though accurate, the maxpooling and spatial pooling operations in the CNN render little interpretability in the recognition model. Our work makes a contribution by providing good accuracy and easy interpretability, in addition to learning efficiency.

VI. Conclusion

This paper addresses handwritten Chinese character recognition. We propose the *MetChar* algorithm for distance metric learning, and the *HybridSelection* algorithm to select distance components. By intelligently learning to combine base components, the learned distance metric has the desired interpretability, learning efficiency and recognition accuracy. Experiments on benchmark data reveal accuracy $\sim 81\%$. Further, we gain insights into the components most useful in Chinese character recognition, through this empirical study. In the future, would investigate other optimization procedures to refine weight assignment for base components. We can consider the use of commonsense knowledge in learning [25] and investigate app development [26] based on related works. We plan to design classification techniques based on attentional neural networks.

Acknowledgement

This work incurs partial support through: startup funds for Dr. Boxiang Dong; a doctoral faculty program for Dr. Aparna Varde; and mentoring inputs from the NSF LSAMP grant for Danilo Stevanovic.

References

[1] "Real-time recognition of handwritten chinese characters spanning a large inventory of 30,000 characters." https://machinelearning.apple.com/2017/09/12/handwriting.html, 2017.

[2] S. Chopra, R. Hadsell, Y. LeCun *et al.*, "Learning a similarity metric discriminatively, with application to face verification," in *CVPR (1)*, 2005, pp. 539–546.

[3] A. Bellet, A. Habrard, and M. Sebban, "A survey on metric learning for feature vectors and structured data," *arXiv:1306.6709*, 2013.

[4] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov, "Neighborhood components analysis," in *NIPS*, 2004, pp. 513–520.

[5] F. Yin and C. L. Liu, "Handwritten chinese text line segmentation by clustering with distance metric learning," *Pattern Recognition*, vol. 42, no. 12, pp. 3146–3157, 2009.

[6] Q. F. Wang, F. Yin, and C. L. Liu, "Handwritten chinese text recognition by integrating multiple contexts," *IEEE Trans. on Pattern Anal. and Machine Intelligence*, vol. 34, no. 8, pp. 1469–1481, 2011.

[7] T. H. Su, T.-W. Zhang, D. J. Guan, and H. J. Huang, "Off-line recognition of realistic chinese handwriting using segmentation-free strategy," *Pattern Recognition*, vol. 42, no. 1, pp. 167–182, 2009.

[8] Q. F. Wang, F. Yin, and C. L. Liu, "Integrating language model in handwritten chinese text recognition," in *IEEE Intl. Conf. on Document Anal. and Recognition*, 2009, pp. 1036–1040.

[9] N. Li and L. Jin, "A bayesian-based probabilistic model for unconstrained handwritten offline chinese text line recognition," in *IEEE Intl. Conf. on Systems, Man and Cybernetics*, 2010, pp. 3664–3668.

[10] H. Nishimura and T. Timikawa, "Offline character recognition using online character writing information," in *IEEE Intl. Conf. on Document Anal. and Recognition*, 2003, pp. 168–172.

[11] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[12] T. Wang, J. Huan, and B. Li, "Data dropout: Optimizing training data for convolutional neural networks," in *IEEE ICTAI*, 2018, pp. 39–46.

[13] A. S. Varde, E. A. Rundensteiner, G. Javidi, E. Sheybani, and J. Liang, "Learning the relative importance of features in image data," in *IEEE ICDE DBrank workshop*, 2007, pp. 237–244.

[14] A. S. Varde, E. A. Rundensteiner, C. Ruiz, D. Brown, M. Maniruzzaman, and R. D. Sisson Jr, "Effectiveness of domain-specific cluster representatives for graphical plots," in *ACM SIGMOD IQIS workshop*, 2006, pp. 24–29.

[15] A. Varde, S. Bique, and D. Brown, "Distance metric learning by greedy, exhaustive and hybrid approaches," Tech. Rep, Virginia State University, VA, 2007.

[16] A. Varde, S. Bique, E. Rundensteiner, D. Brown, J. Liang, R. Sisson, E. Sheybani, and B. Sayre, "Component selection to optimize distance function learning in complex scientific data sets," in *International Conference on Database and Expert Systems Applications*. Springer, 2008, pp. 269–282.

[17] P. Carbonetto, N. De Freitas, and K. Barnard, "A statistical model for general contextual object recognition," in *European Conf. on Computer Vision*. Springer, 2004, pp. 350–362.

[18] M. Z. Hossain, M. A. Amin, and H. Yan, "Rapid feature extraction for optical character recognition," *arXiv:1206.0238*, 2012.

[19] C. Shen, J. Kim, L. Wang, and A. Hengel, "Positive semidefinite metric learning using boosting-like algorithms," *JMLR*, vol. 13, no. Apr, pp. 1007–1036, 2012.

[20] Q. Qian, J. Hu, R. Jin, J. Pei, and S. Zhu, "Distance metric learning using dropout: a structured regularization approach," in *ACM KDD*, 2014, pp. 323–332.

[21] C. C. Chang, "A boosting approach for supervised mahalanobis distance metric learning," *Pattern Recognition*, vol. 45, no. 2, pp. 844–862, 2012.

[22] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *JMLR*, vol. 10, no. Feb, pp. 207–244, 2009.

[23] C. L. Liu, "Normalization-cooperated gradient feature extraction for handwritten character recognition," *IEEE Trans. on Pattern Anal. and Machine Intelligence*, vol. 29, no. 8, pp. 1465–1469, 2007.

[24] Y. C. Wu, F. Yin, and C. L. Liu, "Improving handwritten chinese text recognition using neural network language models and convolutional neural network shape models," *Pattern Recognition*, vol. 65, pp. 251–264, 2017.

[25] N. Tandon, A. Varde, and G. de Melo, "Commonsense knowledge in machine intelligence," *ACM SIGMOD Record*, vol. 46, no. 4, pp. 49–52, 2017.

[26] P. Basavaraju and A. Varde, "Supervised learning techniques in mobile device apps for androids," *ACM SIGKDD Explorations*, vol. 18, no. 2, pp. 18–29.