Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 4059784, 7 pages
https://doi.org/10.1155/2021/4059784

Research Article

Basic Research on Ancient Bai Character Recognition Based on Mobile APP

Zeqing Zhang,[1,2] **Cuihua Lee**,[1] **Zuodong Gao**,[1] **and Xiaofan Li**[1]

[1]*Xiamen University, Amoy, China*
[2]*West Yunnan University of Applied Sciences, Dali, China*

Correspondence should be addressed to Zeqing Zhang; 313460472@qq.com

Received 8 September 2021; Revised 9 November 2021; Accepted 6 December 2021; Published 31 December 2021

Academic Editor: Xingsi Xue

Bai nationality has a long history and has its own language. Limited by the fact that there are fewer and fewer people who know the Bai language, the literature and culture of the Bai nationality begin to lose rapidly. In order to make the people who do not understand Bai characters can also read the ancient books of Bai nationality, this paper is based on the research of high-precision single character recognition model of Bai characters. First, with the help of Bai culture lovers and related scholars, we have constructed a data set of Bai characters, but limited by the need of expert knowledge, so the data set is limited in size. As a result, deep learning models with the nature of data hunger cannot get an ideal accuracy. In order to solve this issue, we propose to use the Chinese data set which also belongs to Sino-Tibetan language family to improve the recognition accuracy of Bai characters through transfer learning. In addition, we propose four transfer learning approaches: Direct Knowledge Transfer (DKT), Indirect Knowledge Transfer (IKT), Self-coding Knowledge Transfer (SCKT), and Self-supervised Knowledge Transfer (SSKT). Experiments show that our approaches greatly improve the recognition accuracy of Bai characters.

1. Introduction

Bai nationality has a long history, splendid culture, and a population of more than one million. Most of them live in Dali Bai Autonomous Prefecture of Yunnan, and the rest are distributed in all parts of Yunnan, Bijie Prefecture of Guizhou, Liangshan Prefecture of Sichuan, Sangzhi County of Hunan, etc. They have their own unique language. Bai language is not only the common communication language of Bai people, but also an important link to condense national emotion and an important carrier of Bai culture development. As the vocabulary, pronunciation and grammar of Bai characters are all Chinese and Tibeto Burmese. The language structure of Bai characters has very important academic value and has been widely concerned by Chinese and national language circles at home and abroad for a long time. For the Bai nationality, whose literature is extremely scarce, its historical and cultural value is self-evident.

In order to promote Bai culture, it is important that people who do not understand the Bai characters can also read the historical documents of Bai nationality or Bai characters on stone steles. It is urgent to study and proposed an automatic model that can recognize the single Bai characters. In this way, once we meet an unknown Bai character, we can take a picture, then use the model recognition, and finally give the explanation of the word.

With the renaissance of neural networks and deep learning, tremendous breakthroughs have been achieved on various recognition tasks [1–6]. Therefore, we consider using deep learning models to do the word recognition of Bai characters. First, we construct a single word data set of Bai characters, which is a handwritten data set by Bai people and culture researches. Due to the requirement for specialized knowledge, the cost of constructing and labeling this data set is high.

Given the data set, we directly train traditional and recent deep learning classification models [1, 7, 8] on this data set, but we find that their performance is not ideal. This is because the success of depth models needs a lot of data support, and the data-hunger nature of depth models leads to their poor performance when there is less data. Because of the need of a lot of expert knowledge, our data set cannot be constructed as large as the traditional classification data set [9–11].

In order to solve this problem, we find that Chinese and Bai language have a high degree of similarity, both belong to the Sino-Tibetan language family (see Figure 1). Therefore, we propose that we can use the way of knowledge transfer [12–15] to transfer a large amount of Chinese knowledge to Bai language, so as to obtain a better accuracy in Bai language. We have designed four methods of knowledge transfer: Direct Knowledge Transfer (DKT), Indirect Knowledge Transfer (IKT), Self-coding Knowledge Transfer (SCKT), and Self-supervised Knowledge Transfer (SSKT).

DKT is a classic idea of knowledge transfer. First, the model is pretrained on the Chinese character data set, then the feature extraction module of the model is used as the parameter initialization of the Bai character recognition network, and finally, the Bai character recognition network is fine-tuned on the Bai character data set. The advantage of this method is that the idea is direct and the implementation is simple, but the disadvantage of this method is also very obvious, that is, the Chinese character label does not contain any semantic information, and it is difficult to guarantee how much knowledge extracted by this hard label can be transferred to Bai characters.

IKT is a method to ensure the quality of knowledge transfer. It is noted that both Chinese and Bai language are composed of 32 basic strokes, just like English is composed of 26 letters. Therefore, the number of basic strokes of each Chinese character is counted, and the number of basic strokes is used as a soft label to train the network. In this way, the network can directly mine the common knowledge of Chinese and Bai language, instead of mining the knowledge through a classification task, so that the knowledge mined by the network can be better transferred to the Bai language.

SCKT and SSKT are two unsupervised knowledge transfer methods. The unlabeled data set is easier to obtain and has lower cost, so the unsupervised knowledge transfer method will be more practical. SCKT is to train a self-encoder [16–18] with Chinese data set and then use the encoded part as the feature extraction part of Bai character recognition network. SSKT uses the method of comparative learning [19–21] to let the data automatically mine the potential knowledge. The biggest advantage of these two methods is that no tags are needed for Chinese data sets, but the disadvantage is that it is difficult to guarantee how much knowledge acquired by these unsupervised methods can be used for knowledge transfer. It is also found that the accuracy of unsupervised knowledge transfer is lower than that of supervised knowledge transfer.

Our main contributions are fourfold: (1) We build a Bai character data set. (2) We propose four methods of knowledge transfer, DTK, ITK, SCTK, and SSTK, to transfer the knowledge of Chinese characters to Bai characters. (3) The four methods proposed in this paper have greatly improved the recognition accuracy of Bai characters. (4) The research could benefit the development of a mobile APP for recognition of Bai characters.

2. Materials and Methods

2.1. Notations. In order to improve the recognition performance of the model, we consider using transfer learning [12–15, 22]. Transfer learning is an ability of a system to recognize and apply knowledge and skills learned in previous domains/tasks to novel domains/tasks. Specifically, let the domain be denoted as $D = \{\mathcal{X}, P(X)\}$, where $\mathcal{X}$ represents the feature space, $P(X)$ represents the marginal probability distribution, and $X \in \mathcal{X}$. And we can define the task as $T = \{\mathcal{Y}, f(\cdot)\}$, where $\mathcal{Y}$ represents the label space and $f(\cdot)$ represents the target prediction function.

The main problem of this paper is how to use the knowledge of Chinese characters to improve the recognition ability of Bai characters. Obviously, the novel domain is composed of Bai characters, which can be defined as $D^b = \{\mathcal{X}^b, P(X^b)\}$. The novel task is also certain, that is, to predict the label of Bai characters. We define the novel task as $T^b = \{\mathcal{Y}^b, f^b(\cdot)\}$ where $f^b : \mathcal{X}^b \longrightarrow \mathcal{Y}^b$. Similarly, the previous domain is composed of Chinese characters, which is defined as $D^c = \{\mathcal{X}^c, P(X^c)\}$. And the design of the previous task $T^c = \{\mathcal{Y}^c, f^c(\cdot)\}$ directly determines the quality of the knowledge extracted from Chinese characters. The design of the previous task is also the focus of this paper. Finally, in this paper, we give the design of four kinds of previous tasks.

2.2. Transfer Learning Approaches. We divide transfer learning approaches into supervised and unsupervised.

Two approaches were designed as supervised, Direct Knowledge Transfer (DKT) and Indirect Knowledge Transfer (IKT); DKT directly uses Chinese character label as the training task while IKT uses common Chinese and Bai character attributes as a knowledge transfer bridge. In addition, we also designed two unsupervised transfer learning approaches. One is Self-coding Knowledge Transfer (SCKT). As the name suggests, this method uses self-encoder [16–18] to extract low-frequency information of characters, that is, commonness. The other is Self-supervised Knowledge Transfer (SSKT). Thanks to the prosperity of self-supervised [23–26], self-supervised proposes a method that allows data to monitor themselves to extract features. Contrastive learning [19–21] is a promising way effectively extracts features used to distinguish different categories from data.

The details of these four approaches are as follows.

Direct Knowledge Transfer. The idea and implementation of this approach is very direct. Because the target task T^b is a classification, the most intuitive way is to design the source task also as a classification problem. That is, $T^c = \{\mathcal{Y}^c, f^c(\cdot)\}$ where $\mathcal{Y}^c$ represents the label space of Chinese characters and $f^c : \mathcal{X}^c \longrightarrow \mathcal{Y}^c$. Suppose $P = f(X^c)$, where $P = [p_1, p_2, \cdots, p_n]$ represents the probability that an example is classified into each of the n possible Chinese characters. Then, the training loss of this approach using source domain T^c using previous domain D^c is as follows:

$$\mathcal{L} = -\log \frac{\exp(p_k)}{\sum_{i=1}^{i=n} \exp(p_i)}, \tag{1}$$

where p_k represents the the probability of the correct label.

The advantage of this approach is that it is straightforward with a simple implementation. The fundamental purpose of this approach is to separate Chinese characters,

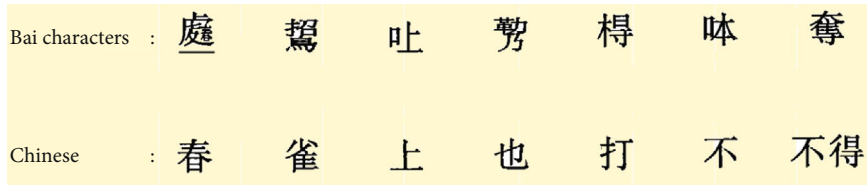


Figure 1: Comparison between Chinese characters and Bai characters.

so there is a part of the method focused on learning the differences between Chinese characters. Although the characteristics between Bai and Chinese characters are different, knowledge learned in a task can be transferred to the target task.

Indirect Knowledge Transfer. Since Chinese character label does not contain any semantic information, we design a label containing semantic information. It is observed that Chinese and Bai characters are composed of 32 basic strokes, as shown in Figure 2. Intuitively, we can use these 32 strokes as soft labels to better transfer the knowledge of Chinese characters to Bai characters. That is, $T^c = \{\mathcal{Y}^c, f^c(\cdot)\}$ where $\mathcal{Y}^c$ represents the label space of 32 basic strokes. If $Y == [y_1, y_2, \cdots, y_{32}] \in \mathcal{Y}^c$, then Y is a vector with a length of 32, and y_i represents the number of the i-th basic strokes. Then, the loss of training this previous task T^c using previous domain D^c is as follows:

$$\mathcal{L} = \|Y - f(X)\|_2^2. \tag{2}$$

The advantage of this approach is that all labels contain rich semantic information, which is shared by Chinese and Bai characters. In this way, the knowledge of 32 basic strokes extracted from Chinese characters can be transferred to Bai characters. The disadvantage is that we need to label each Chinese character with 32 basic strokes, which requires a certain amount of extra work.

Self-coding Knowledge Transfer. In fact, the above two approaches need annotated Chinese characters, where annotation inevitably brings a lot of work. Since a lot of unlabeled examples of Chinese characters are available, a natural idea is knowledge from unlabeled Chinese characters. First, we consider the classical unsupervised learning method: self-encoder, which can learn to compress high-dimensional data into low dimensional without losing information as much as possible. That is, $T^c = \{\mathcal{Y}^c, f^c(\cdot)\}$ where $\mathcal{Y}^c = \mathcal{X}^c$ and $f^c(\cdot)$ is a structure that first encodes and compresses the data and then decodes and restores the data. Then, the loss of training of this approach T^c using previous domain D^c is as follows:

$$\mathcal{L} = \|X - f(X)\|_2^2. \tag{3}$$

However, there is no guarantee that low-frequency information is the effective knowledge to aid Bai character recognition. This task is similar to word2vec [27]; in low latitude space, similar words will still be close together, so it is still an effective method.

Self-supervised Knowledge Transfer. Self-supervised learning [23–26] has gained great attention in recent years because it can automatically extract knowledge in data. Among them, contrastive learning [19–21] has made surprising progress. Therefore, using the recent comparative learning method MoCo [28] to automatically extract the knowledge of Chinese characters has become a natural choice. That is, $T^c = \{\mathcal{Y}^c, f^c(\cdot)\}$ where $\mathcal{Y}^c = [k^+, k_1^-, k_2^-, \cdots, k_n^-]$. k^+ represents that the positive sample used in contrastive learning is usually obtained from the same picture using different data expansion methods, and the others represent negative samples, which are obtained from other pictures. Then, the loss of training approach T^c using source domain D^c is as follows:

$$\mathcal{L} = -\log \frac{\exp\left(X \cdot k^+\right)}{\exp\left(X \cdot k^+\right) + \sum_{i=1}^{i=n} \exp\left(X \cdot k_i^-\right)}. \tag{4}$$

The features extracted by MoCo [28] have achieved encouraging results in many tasks, such as image classification [9] and target detection [29]. The advantage of this approach is that it can inherit this powerful feature extraction ability. However, the training of comparative learning needs larger batch size, which has higher requirements for hardware cost. At the same time, it is difficult to guarantee the quality of the knowledge extracted by comparative learning, which can only be verified by downstream tasks.

2.3. Model Training. After learning a model in the source domain, the known can be transferred to the target domain 2.3.

3. Experimental Results

3.1. Experimental Setup

3.1.1. Data Sets. We build a large data set with 400 Bai characters. Because there is a certain overlap between Bai characters and Chinese characters, we only select Bai characters which are quite different from Chinese ones to compose and build this data set. There are about 2,000 samples for each word and character, all written by Bai people and Bai culture lovers. We split the data set into 50. In addition, we also collected a large data set of Chinese characters. The data set consists of 509 Chinese characters, each of which has about 50 samples. In order to Indirect Knowledge Transfer, we also label each Chinese character with 32 strokes.

3.1.2. Evaluation Protocol. We evaluate the proposed method in terms of the average per-class Top-1 accuracy (ACC).

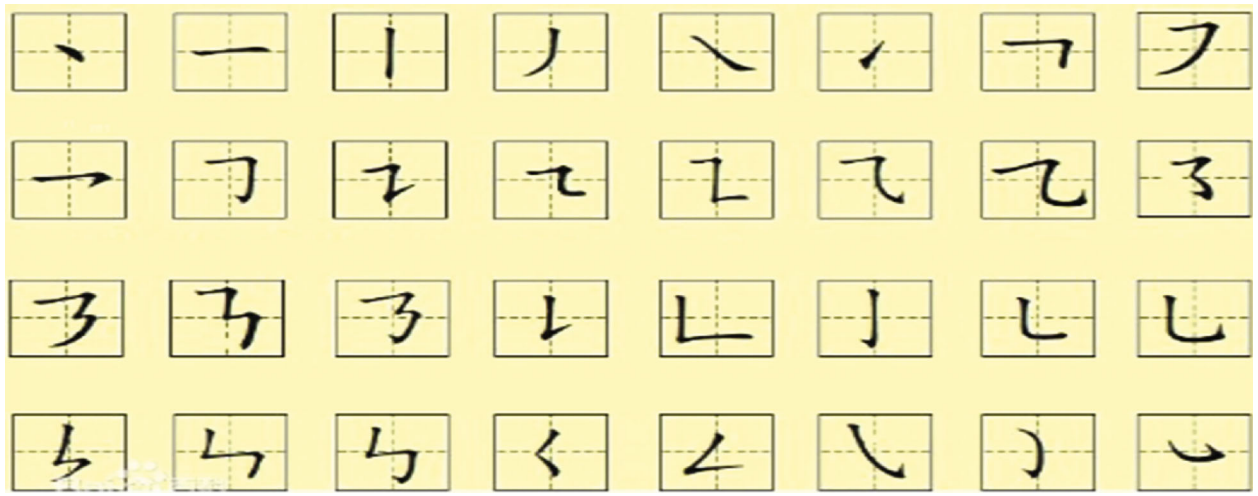


Figure 2: 32 basic strokes of Chinese characters and Bai characters.

Require: Chinese character data set D^c and Bai character data set D^b.
Ensure: Function $f^b(\cdot)$ for Bai character classification.
Initialize the parameters of both $f^c(\cdot)$ and $f^b(\cdot)$.
1: while the $f^c(\cdot)$ does not converge do
2: for samples in D^c do
3: Optimize $f^c(\cdot)$ by Eq.(1) or Eq.(2) or Eq.(3) or Eq.(4).
4: end for
5: end while
6: The parameters of feature extraction part in $f^b(\cdot)$ are replaced by those in $f^c(\cdot)$.
7: while the $f^b(\cdot)$ does not converge do
8: for samples in D^b
9: Optimize $f^b(\cdot)$ by Cross entropy loss.
10: end for
11: end while

Algorithm 1: Proposed approach.

3.1.3. Classification Model. We use three classical classification models: AlexNet [8], VGG19 [7], and ResNet101 [1]. Comparing multiple models, we can analyze that our method is effective and has strong generalization ability.

3.1.4. Implementation Details. We use SGD optimizer with learning rate (lr) = 0.01 and a batch size of 64 to train DKT, IKT, and SCKT. On the other hand, we use SGD optimizer with lr = 0.001 and a batch size of 512 to train SSKT. Finally, we use SGD optimizer with lr = 0.01 and a batch size of 64 to train $f^b(\cdot)$. We use StepLR learning rate adjustment strategy, where the learning rate every 20 epochs becomes 0.8 of the original.

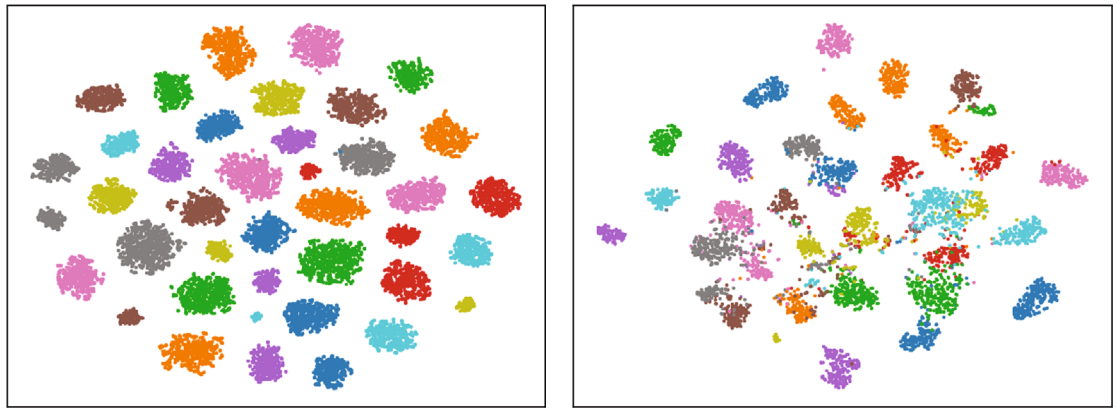
3.2. Accuracy Analysis. The accuracy comparison of the transfer learning approaches is shown in Table 1.

We observe that the proposed IDK achieves significant improvements over the other approaches. On the three CNN models, the accuracy is 12.01%, 9.56%, and 10.00% higher than that without transfer learning. At the same time, this training strategy of transfer learning is the most accurate of the four approaches we proposed. This fully shows that our design of 32 basic strokes as soft labels can transfer the knowledge learned from Chinese characters to Bai characters. Although the 32 basic stroke label does not consider the structure and position, it is still a very effective means of transferring learning based on results.

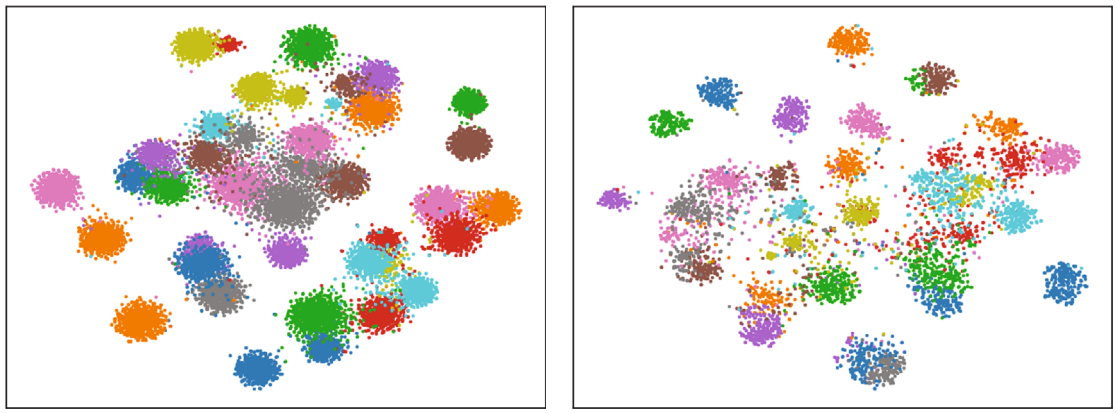
Table 1: Accuracy comparison of different CNNs and transfer learning approaches. No means using Bai characters to train the model directly, and transfer learning is not used.

CNN	No	DKT	IKT	SCKT	SSKT
AlexNet	73.16	83.42	85.17	74.82	77.94
VGG19	78.28	82.92	87.84	78.63	80.21
ResNet101	78.54	87.82	88.54	80.41	82.09

The second high accuracy was obtained by DKT. It uses hard labels directly, that is, labels for each word, to pretrain the models. Although this label does not contain any semantic information, the model still extracts relevant features that



(a) IKT (b) DKT



(c) SCKT (d) SSKT

Figure 3: Different methods of feature visualization results.

can be used for knowledge transfer in the pretraining. On the three models, the accuracy is 10.26%, 4.64%, and 9.28% higher than that without transfer learning. The results showed that this approach can also bring a good improvement in the accuracy, but there is a certain lack of interpretability of the knowledge transferred to the target task.

The accuracy of unsupervised transfer learning approaches SCKT and SSKT is lower than that of supervised approaches. However, it is also an excellent solution if the data set is difficult to get annotation. The essence of SCKT is to seek a low latitude compression of data. Obviously, most of the knowledge used for compression is not directly transferable to another task, so the improvement in accuracy is not obvious. On the three models, the accuracy is 1.66%, 0.35%, and 1.87% higher than that without transfer learning. In fact, compared with the Bai character training model directly, the accuracy is not greatly improved.

Although SSKT cannot be compared to supervised approaches, it was significantly better than SCKT. On the three models, the accuracy is 4.78%, 1.93%, and 3.55% higher than that without transfer learning. Although the latest comparative learning method has been able to compare with the full supervision method, it needs a huge data set to bring. This is also the reason why our SSKT method is inferior to the full supervision method. In many cases, it is difficult for us to obtain a large number of unlabeled data. At that time, this method is the most suitable. Of course, this method requires additional calculation cost for hardware, and it is also a disadvantage that cannot be ignored.

3.3. Feature Visualization Analysis. In order to further illustrate the effect of these four approaches, we directly use the pretrained network to extract the features of 40 Bai characters. Then, t-SNE [30] algorithm is used to visualize these features, as shown in Figure 3. It can be seen that, after pretraining with the IKT, the extracted features can be well distinguished even if there is no fine-tuning in the Bai character data set. Although the features extracted by DKT method have a good degree of aggregation within classes, there is some overlap between classes. Although some of the features extracted by SCKT and SSKT can be well distinguished, most of them will overlap each other. Through the visualization results, we have a more intuitive understanding of the differences between the four methods, and also explain why IKT method is better than other methods.

3.4. Convergence Speed Analysis. In Figure 4, we show the difference of convergence speed of different CNNs. It can be observed that the convergence speed of the CNNs is greatly improved after the application of the knowledge transfer methods. Without the use of knowledge transfer, the CNNs will go through multiple epochs before it starts to converge. However, after the use of the knowledge transfer method, the CNNs will converge from the beginning of

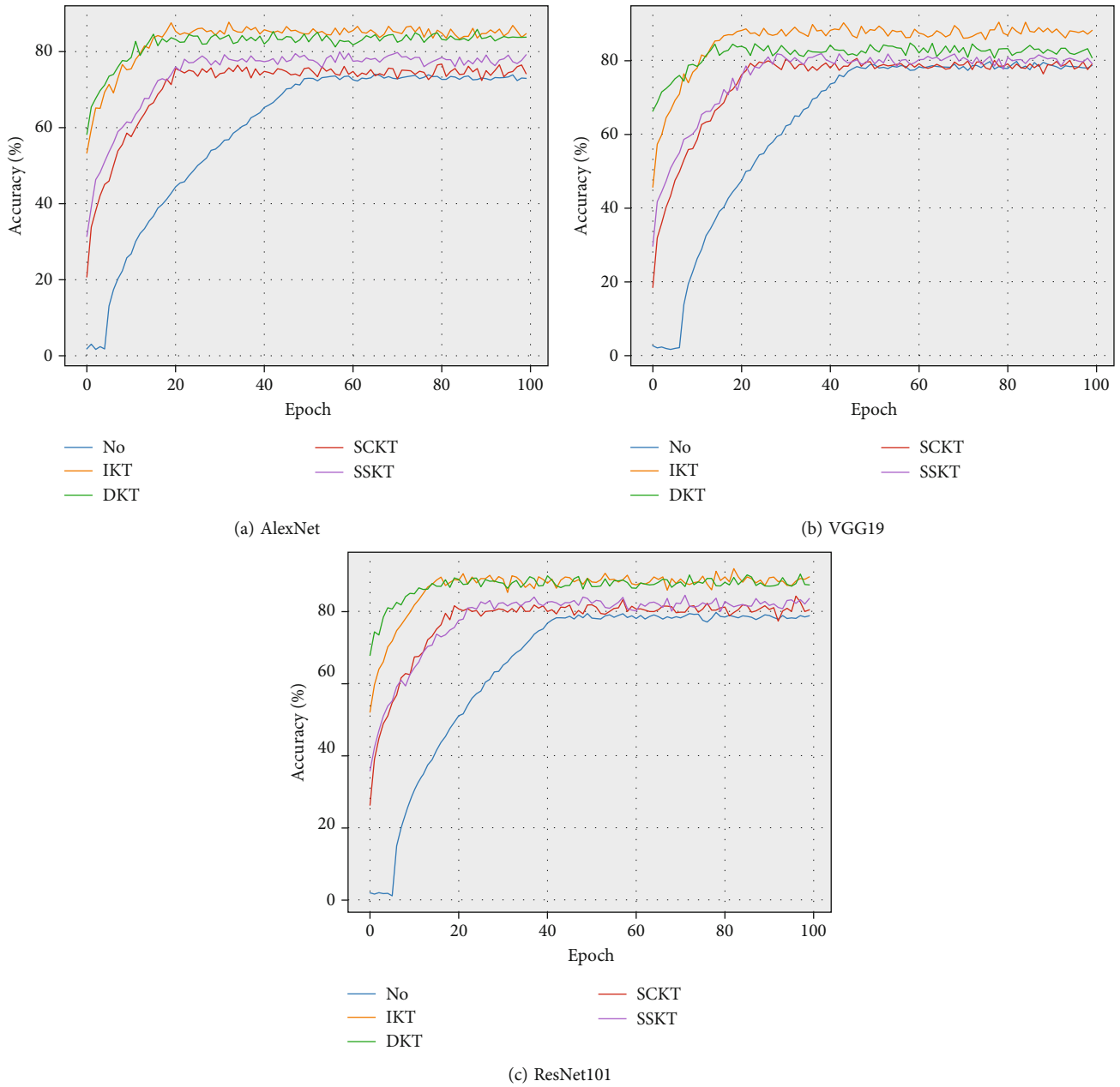


Figure 4: Convergence speed of different methods.

training. This is because the CNN has recognize Bai characters. In other words, the CNN has a good initial parameter, so it can converge quickly.

4. Conclusion

Bai nationality, as a nation with a long history in China, not only has its own language but also has created brilliant culture. However, with the development of the times, because fewer and fewer people know Bai characters, Bai culture is dying out. In order to make people who love Bai culture and related researchers can read Bai literature smoothly, this paper mainly studies how to train a high-precision Bai character recognition CNNs. First, we build a data set of Bai characters, but limited by the need of expert knowledge, so the data set is limited in size. As a result, those depth models that need a lot of data-driven cannot achieve satisfactory results on this data set. In order to solve this problem, we propose to use the Chinese data set which also belongs to Sino-Tibetan language family to help improve the recognition accuracy of Bai characters through knowledge transfer. In addition, we propose four methods of knowledge transfer: Direct Knowledge Transfer (DKT), Indirect Knowledge Transfer (IKT), Self-coding Knowledge Transfer (SCKT), and Self-supervised Knowledge Transfer (SSKT). Sufficient experiments not only show that our method can greatly improve the recognition accuracy of Bai characters but also show the advantages of our method from the visualization and convergence speed.

Data Availability

We build a large data set of Bai characters. There are a total of 400 Bai characters. Because there is a certain overlap between Bai characters and Chinese characters, we only select Bai characters which are quite different from Chinese characters to build this data set. There are about 2,000 samples for each word, all written by Bai people and Bai culture lovers. The training set and the test set are about one to one. At present, the data set is still private and will be made public later.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, https://arxiv.org/abs/1512.03385.

[2] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: deep learning on point sets for 3d classification and segmentation," 2016, https://arxiv.org/abs/1612.00593.

[3] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," 2015, https://arxiv.org/abs/1506.01497.

[4] X. Xingsi, W. Xiaojing, J. Chao, M. Guojun, and Z. Hai, "Integrating sensor ontologies with global and local alignment extractions," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6625184, 10 pages, 2021.

[5] X. Xue and J. Zhang, "Matching large-scale biomedical ontologies with central concept based partitioning algorithm and adaptive compact evolutionary algorithm," *Applied Soft Computing*, vol. 106, article 107343, 2021.

[6] X. Xue and J. Chen, "Matching biomedical ontologies through compact differential evolution algorithm with compact adaption schemes on control parameters," *Neurocomputing*, vol. 458, pp. 526–534, 2021.

[7] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very deep convolutional networks for natural language processing," 2016, https://arxiv.org/abs/1606.01781.

[8] S. Targ, D. Almeida, and K. Lyman, "Resnet in resnet: Generalizing residual architectures.," 2016, CoRR abs/1603.08029.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems 25*, pp. 1106–1114, 2012.

[10] G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, "Emnist: an extension of mnist to handwritten letters," 2017, https://arxiv.org/abs/1702.05373.

[11] M. Swofford, "Image completion on CIFAR-10," 2018, https://arxiv.org/abs/1810.03213.

[12] M. Chen, Z. E. Xu, K. Q. Weinberger, and F. Sha, "Marginalized denoising autoencoders for domain adaptation," 2012, https://arxiv.org/abs/1206.4683.

[13] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: a deep learning approach," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 513–520, Bellevue, Washington, USA, 2011.

[14] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *CVPR*, pp. 1717–1724, IEEE Computer Society, 2014.

[15] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., pp. 3320–3328, 2014.

[16] Y. J. Fan, "Autoencoder node saliency: selecting relevant latent representations," 2017, https://arxiv.org/abs/1711.07871.

[17] H. Ishfaq, A. Hoogi, and D. Rubin, "TVAE: triplet-based variational autoencoder using metric learning," 2018, https://arxiv.org/abs/1802.04403.

[18] Q. Li, X. Zheng, and X. Wu, "Collaborative autoencoder for recommender systems," 2017, https://arxiv.org/abs/1712.09043.

[19] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*, pp. 1597–1607, 2020.

[20] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, https://arxiv.org/abs/1807.03748.

[21] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *ECCV (11). Lecture Notes in Computer Science, vol. 12356*, A. Vedaldi, H. Bischof, T. Brox, and J. M. Frahm, Eds., pp. 776–794, Springer, 2020.

[22] S. Mirsamadi and J. H. L. Hansen, "Multi-domain adversarial training of neural network acoustic models for distant speech recognition," *Speech Communication*, vol. 106, pp. 21–30, 2019.

[23] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," 2018, https://arxiv.org/abs/1808.06670.

[24] I. Misra, C. L. Zitnick, and M. Hebert, "Shuffle and learn: unsupervised learning using temporal order verification," in *ECCV (1). Lecture Notes in Computer Science, vol. 9905*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., pp. 527–544, Springer, 2016.

[25] J. Wu, X. Wang, and W. Y. Wang, "Self-supervised dialogue learning," in *ACL (1)*, A. Korhonen, D. R. Traum, and L. Márquez, Eds., pp. 3857–3867, Association for Computational Linguistics, 2019.

[26] Z. Wu, Y. Xiong, S. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance-level discrimination," 2018, https://arxiv.org/abs/1805.01978.

[27] X. Rong, "word2vec parameter learning explained," 2014, https://arxiv.org/abs/1411.2738.

[28] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9726–9735, 2020.

[29] P. R. P. A. P. S. T. Vinod, "Object detection an overview," *International Journal of Trend in Scienti_c Research and Development*, vol. 3, no. 3, pp. 1663–1665, 2019.

[30] N. Rogovschi, J. Kitazono, N. Grozavu, T. Omori, and S. Ozawa, "T-distributed stochastic neighbor embedding spectral clustering," in *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 1628–1632, Anchorage, AK, USA, 2017.