COMP9444 Project Summary

Fashion Item Classification using Deep Learning

Chenyan Li(z5467843) Xiaojie Yu(z5469466) Cisco Xing(z5522418)
Jiacheng Ling(z5529848) Wenjia Du(z5536121)

I. Introduction

The classification of clothing styles is a recent topic in computer vision research and has many interesting applications[1], including e-commerce, criminal law, and online advertising. Meanwhile, automated classification of fashion items is crucial for various applications in the fashion industry, such as product categorization, recommendations, and search optimization. However, accurately classifying fashion items can be challenging due to the wide variety of styles, designs, and visual similarities between different categories. This project aims to address this challenge on the UT Fashion 100 dataset by leveraging deep learning techniques, particularly convolutional neural networks (CNNs), to develop a robust and efficient system for classifying fashion items based on images.

II. Related Work

Image classification is the process of distinguishing different categories of images based on their semantic information. Traditional image classification methods typically involve two steps: manual feature extraction and classifier-based object category determination. Convolutional Neural Networks (CNNs) are one of the primary models used for image classification in deep learning. Since the creation of CNN, it has played an important role in image classification. CNNs are particularly prominent in image classification. Many CNN-based algorithms have been proposed and widely used in image classification tasks, including AlexNet, VGGNet, GoogleNet, ResNet and LeNet.[2] Also, some researchers try to use a TinyVGG which based on VGGNet to extract feature.[3] And with optimizer functions such as SGD, which is more effective to minimize the training loss and improve the model accuracy during training.

Although CNN have become a major model in computer vision applications. Their success is partly due to better performance from training on large data sets, but this requires large amounts of labeled data and is expensive, especially if manual labeling is required (as in this project). Semi-supervised learning[4] (SSL) and transfer learning[5] provides a solution to alleviate the need for labeled data by combining a small amount of labeled data with a large amount of unlabeled data. The SSL approach utilizes unlabeled data to significantly reduce labeling costs and improve model performance. Transfer learning focus more on take advantages on a bigger well-trained model.

III. Methods

To complete the task, three general approaches are chosen and performed. We use direct learning, transfer learning and semi-supervised learning.

Since the biggest problem we are facing in this task is the insufficient labeling in the dataset, some specialized methods are necessary. But it is very intuitive to simply use some models to do the classification directly, as a contrast with other methods. Thus, for direct learning, we choose two pre-trained models, ResNet and VGG, together with a 3-convolution-layer CNN to compare each of their performance as well as their overall accuracy.

Next, considering the biggest flaw from insufficient labeling is that there are not enough data to properly train the model without causing under-fitting or over-fitting, we turn to transfer learning method. Utilizing another similar fashion item dataset (more details in the next section of this report), we implemented three models by ourselves, MiniVGG, DenseNet-40 and ResNet-18. Then we choose the best performance model, MiniVGG, and modified it into a hierarchy CNN that can output not only the eight sub-category classification result, but also the four-category classification result at the same time.

For the last method, we choose semi-supervised learning method which is especially useful for our A100 dataset. In this work, we implemented the FixMatch method, a semi-supervised learning technique that combines consistency regularization and pseudo-labeling to enhance model performance, which effectively utilizes unlabeled data to significantly enhance the classification accuracy of fashion items. We used a WideResNet neural network as the training model for this task.

IV. Experimental Setup

1. Datasets

Fashion Product Images (Small)

URL: https://www.kaggle.com/datasets/paramaggarwal/fashion-product-images-small

This dataset is used for transfer learning and all the categories are organized based on this dataset. It originally has 44441 images of 143 different classes. Considering the relevance and balance between each category, we choose 47 classes into 8 categories. The number of chosen images is in the table below. For H-CNN's 4-category, put 'Sunglasses' and 'Watches' into 'Accessories', and combine 'Dress', 'Pants' and 'Tops' into 'Clothes'.

Category	Accessories	Bags	Dress	Pants	Shoes	Sunglasses	Tops	Watches	Total
number	1248	3307	2899	1921	8306	1073	12607	2542	33603

Clearly, it is not a perfectly balanced dataset. To maintain the balance for the model, we use under-sampling to randomly choose 900 images out of each category, in which 800 for training and 100 for testing. Combining the together gives us a training set of size 6400 and test set of size 800. In order to make full use of all images, we change the images every epoch, using similar ideas as cross validation and bootstrap. In this way the model sees different data in each epoch, thereby increasing data diversity and the model's generalization ability.

The UT Fashion dataset (A100 dataset)

This dataset is the target dataset given for this project. It contains two sets of images, LAT and AAT. The LAT folder has 7427 images with only 680 data labels. The AAT folder has 988 images with 983 data labels. The dataset in total contains 8415 images with 1663 labeled images. We re-organize the labels into 8 categories and combine the two folders labeled images together, we get the data as in the table below.

We implement stratified sampling, which ensure that class distributions are preserved and thereby improving the accuracy and stability of model performance evaluation, to construct our datasets while preventing class imbalance.

Category	Accessories	Bags	Dress	Pants	Shoes	Sunglasses	Tops	Watches
number	371	289	215	128	314	7	296	43

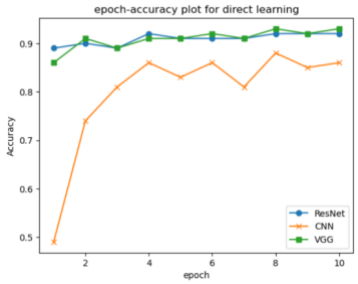
2. General setup

For direct classification and transfer learning, we choose SGD as optimizer. The learning rate is 0.01 and momentum is 0.5. The batch size is 64. We use accuracy and confusion matrix as evaluation for the models. Direct classification uses the labeled A100 images to train and test, and transfer learning uses the selected Fashion Product Images first to do feature learning, then transfer into labeled A100 to train and test.

For our work of semi-supervised learning, we choose WideResNet as our training model. We implement a cosine schedule with warmup method to dynamically adjust learning rate. We set the confidence threshold as 0.95, initial learning rate as 0.03 and the batch size as 256. With each epoch the model output the accuracy produced on the test dataset, loss values of both labeled data and unlabeled data and confusion matrix.

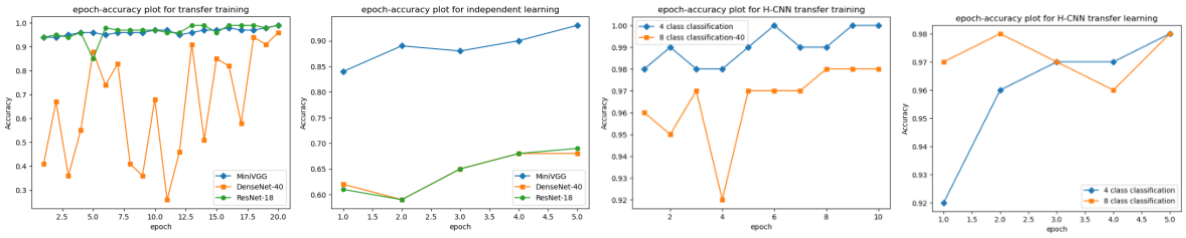
V. **Results**

For direct learning, the pre-trained VGG model achieves the highest accuracy, which is 93.4%, followed by pre-trained Resnet(91.6%), and CNN(86.5%). Only pre-trained models have a good result, the overall performance is not satisfying due to insufficient images for training and imbalanced labeling.



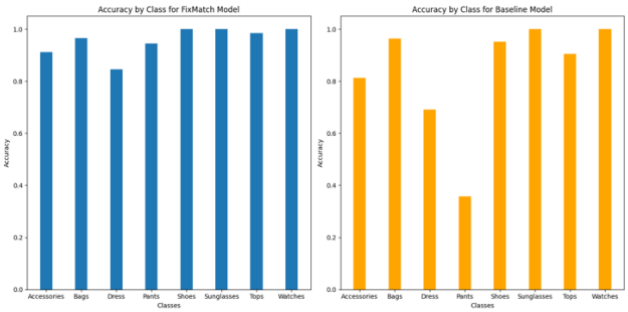
For transfer learning, the three models in feature learning part have an accuracy of 96.5% for DenseNet and 99.3% for MiniVGG and ResNet. In the transfer learning part, MiniVGG achieved 92.7% accuracy, but DenseNet and ResNet only got 68.4% and 68.7%. We suspect the reason for DenseNet to have a jumping accuracy is that there may be some overfitting because every layer is connected with all previous layers. The result also indicates whether there may be overfitting in DenseNet and ResNet for their complex mechanism are designed for much deeper networks.

For H-CNN part, our work achieved very well-performed results. In feature learning, it predicts 4-class labels with 99.8% accuracy and 8-class labels with 97.9%. In the transfer learning, it achieved 97.9% accuracy in both tasks. Thus, this is the best model for this classification task, it is simple and effective with many modification possibilities.



For FixMatch, we use VGG-16 and AlexNet as baseline models, in 8-class and the original 13-class datasets, the result and accuracy by class plot are as follows:

Method	8 classes	13 classes
VGG-16	85.54%	79.43%
AlexNet	56.91%	43.12%
FixMatch (WideResNet)	94.53%	89.94%



As the plot presents, the FixMatch method shows consistently high and similar accuracy across all classes compared to the baseline model.

In the end, there is the final result for all models in all methods we have done. This will give an overall impression and direct comparison for all our work. '-' indicates this model does not produce this output. From the table, of all models, the modified MiniVGG and FixMatch produced the most satisfying results.

method	model	4-class results	8-class results	13-class results
Direct learning	3-layer CNN	_	86.5%	_
	Pre-trained ResNet	_	91.6%	_
	Pre-trained VGG	_	93.4%	_
Transfer learning	MiniVGG	_	92.7%	_
	DenseNet-40	_	68.4%	_
	ResNet-18	_	68.7%	_
	MiniVGG modified H-CNN	**97.9%**	**97.9%**	_
Semi-supervised learning	VGG-16	_	85.54%	79.43%
	AlexNet	_	56.91%	43.12%
	FixMatch	_	**94.53%**	**89.94%**

VI. Conclusions

After comparing the results, we propose two solutions for this project. One is MiniVGG model for transfer learning, the other is FixMatch using semi-supervised learning. The MiniVGG model is proved to be simple and effective, the modified H-CNN achieved the highest accuracy in all methods. FixMatch is specialized in handling massive unlabeled data while maintaining high accuracy.

However, there are still some limitations for our project. We divided ourselves into three groups and work independently on different methods, so the results we get is not that consistence and some models are used multiple times in different methods. Given more time, we will unify our method results to be more comparable and we will try to make full use of the massive unlabeled data in transfer learning to finalize the classification of all images. Even more, we may spend some effort to manually label all the images and test the whole dataset to improve our works.

Reference

[1] Hu, W., Huang, Y., Wei, L., Zhang, F., & Li, H. (2015). Deep convolutional neural networks for hyperspectral image classification Journal of Sensors, Volume 2015, Article ID 258619, 12 pages. http://dx.doi.org/10.1155/2015/258619

[2]Meshkini, K., Platos, J. and Ghassemain, H., 2020. An analysis of convolutional neural network for fashion images classification (fashion-mnist). In Proceedings of the Fourth International Scientific Conference "Intelligent Information Technologies for Industry"(IITI'19) 4 (pp. 85-95). Springer International Publishing.

[3]Xin, J., Yi, T.J., Yi, V.P., Yu, P.J. and Salam, Z.A.A., 2023. Convolutional Neural Network for Fashion Images Classification (Fashion-MNIST). Journal of Applied Technology and Innovation (e-ISSN: 2600-7304), 7(4), p.11.

[4]Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C.A., Cubuk, E.D., Kurakin, A. and Li, C.L., 2020. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. Advances in neural information processing systems, 33, pp.596-608.

[5]Hussain, M., Bird, J.J., Faria, D.R. (2019). A Study on CNN Transfer Learning for Image Classification. In: Lotfi, A., Bouchachia, H., Gegov, A., Langensiepen, C., McGinnity, M. (eds) Advances in Computational Intelligence Systems. UKCI 2018. Advances in Intelligent Systems and Computing, vol 840. Springer, Cham. https://doi.org/10.1007/978-3-319-97982-3_16.