IT1244 Project Report on Clothing Image Classification

By Hu Can, Liao Jingyu, Shen Yuqing, Zhao Xingyue

Abstract

Image classification is a supervised learning problem in Artificial Intelligence that studies the features in an image and identifies their categories. It is popularly used in medical imaging, object recognition in satellite imagery, traffic management systems, machine visions and other applications. In this project, we use four different models including Decision Tree, Random Forest, Artificial Neural Network(ANN) and Convolutional Neural Network(CNN) to categorise clothes into 10 types based on 784 attributes of pixels. The methods were compared based on accuracy and Cohen's Kappa Matric. Through model comparisons, we found out that CNN model achieved the best performance with an accuracy of 0.911 and a Cohen's kappa coefficient of 0.896. In this report, all four models' applications and limitations will be explained and elaborated.

1. Introduction

Image classification and recognition play a pivotal role in crafting a personalised shopping experience in the fashion industry. The integration of image recognition within e-commerce platforms has profoundly streamlined customer searches. For example, the Asos app, a cosmetic retail leader, enables customers to snap a photo of a clothing item, prompting the app to suggest similar products—far more efficient than traditional keyword-based searches. This technology enables brands which support visual search to increase their digital commerce revenue by 30% (Muses, 2019). Consequently, the importance of an efficient image classification and recognition model in enabling robust visual searching cannot be overstated.

Previous research in image classification and recognition has leveraged supervised machine learning algorithms, including Decision Trees, Random Forests, Artificial Neural Networks (ANN), and Convolutional Neural Networks (CNN). These algorithms exhibit distinct advantages and limitations. Decision trees, for instance, tend to overfit the training data, resulting in suboptimal performance when applied to the complete dataset. In contrast, Random Forests employ a technique known as bagging, wherein multiple decision trees are trained on diverse subsets of the training data, reducing the risk of overfitting (R, 2015). Furthermore, the effectiveness of ANN is substantially influenced by the training data's size, with larger datasets generally yielding higher classification accuracy. In contrast, CNN presents a solution to this issue by achieving superior accuracy with smaller sample sizes compared to ANN(Maruyama et al., 2018). In this paper, our objective is to create an image classification model capable of categorising ten distinct clothing types. We will implement and assess the performance of two machine learning algorithms we have studied in the IT1244 module: Artificial Neural Network (ANN) and Convolutional Neural Network (CNN). We will also explore two other algorithms beyond the syllabus: Decision Tree and Random Forest. The evaluation of their performance will be conducted using accuracy and Cohen Kappa Matrix as benchmarks.

2. Dataset

2.1 Sources of dataset

The dataset is the 3.5.1 Clothing Dataset provided in the IT1244 Project Handout. It is a CSV file that contains 10 types of clothes with each type containing 6000 instances. There are 785 columns, in which the first column indicates the categories of clothes and the rest describe the attributes of the instances. No external dataset is included in our project.

2.2 Features of dataset

- The dataset contains 28X28 pixel images, with pixel values already stored in CSV file, eliminating the need for data flattening.
- The data are divided into 10 categories, each containing 6000 instances, ensuring a balanced dataset.
- There are also no missing values.
- As the images are grayscale, meaning that they only have a single layer. This allows the dataset's dimensions to seamlessly align with those of a standard array.

3. Model

Four different classifier algorithms are used to solve the problem, namely Decision Tree, Random Forest, Artificial Neural Network (ANN) and Convolutional Neural Network (CNN).

3.1 Decision Tree

The Decision tree was selected due to its ease of implementation and interpretability. The model, based on a series of if-else conditions, is intuitive to understand and visualise. Moreover, decision trees act as the "baseline" model for random forest which will be introduced later in the report. In Python's decision tree implementation, features with the lowest Gini index will be used as the root node. Subsequent decision nodes will uphold larger Gini indices. Despite being simple, decision trees are prone to overfitting, and the risk of that is exacerbated especially when the datasets are large.

3.2 Random Forest
The Random Forest model was selected as it leverages multiple decision trees to produce a more accurate outcome than the Decision Tree model. Each tree is constructed by repeatedly dividing the data into subsets based on random feature selections and bootstrapped samples. In the end, a diverse range of trees are created and their individual predictions are combined to produce a final result. The diversity of trees attributed to the randomness in feature and data selection reduced the risk of overfitting and increases the robustness of the model.

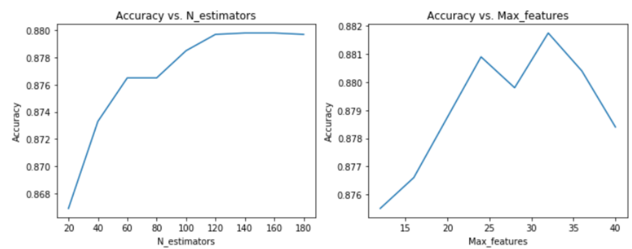
3.3 Artificial Neural Network
The ANN model was chosen as its massive parallelism and fast adaptability ensured its efficiency in all image processing steps (Cristea, 2009). Labelled training pixel values are normalised and passed into the model constructed using Keras, which comprises a flattened input layer, a dropout layer to reduce overfitting, and two dense layers for feature learning and classification. After training, the model is evaluated on the test data. It also plotted accuracy trends over epochs and batch size effects on validation accuracy.

3.4 Convolutional Neural Network
The CNN model was employed considering its inherent strength in image processing. Through the use of image filters, features of the input are accurately extracted from the image resource and recorded under its convolution layers during training. Furthermore, as patterns are learned by the deep learning model in a space-invariant manner, this is exceptionally favourable in our task of object classification, as the program would be required to accurately identify dispositioned features in the test data.

4. Result and Discussion

4.1 Decision Tree
For the decision tree model, accuracy fluctuated between 79 percent and Cohen's Kappa coefficient was around 76.5 percent, which is the lowest among all the four models. Although the decision tree is the simplest to implement, it proved to be computationally expensive, as the dataset contains 784 variables. This complexity led to the calculation of 784 Gini indices. Additionally, decision tree is good at categorial variable dataset. However, the clothing dataset is numerical and hence extra threshold is needed to split the dataset.

4.2 Random Forest
The classification using Random Forest yielded an overall accuracy of 88.2%. To fine-tune the model, experiments were conducted to find the optimal values of the critical hyperparameter: the number of decision trees employed (n_estimators) and the number of features considered in each tree-building step (max_features). A larger value of n_estimators results in a trade-off between higher accuracy and increasing time and space requirements. Hence, the value where the model reached a point of diminishing re urns in accuracy improvement was selected (Figure 1a). A recommended value of max_features for classification tasks is the square root of the total number of features ac cording to empirical research (Breiman, 2001). Thus, values close to the recommended value were investigated to find the optimal value for this project (Figure 1b).



(a) (b)

Figure 1: Graph of accuracy of RT model against n_estimators (a) and max_features (b)

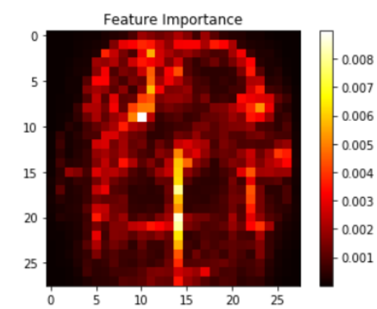


Figure 2: Graph of pixels coloured according to their importance for the classification

In addition to accuracy, the results also provided valuable insights into the importance of individual pixels in the classification process (Figure 2).

4.3 Artificial Neural Network
For the ANN model, the accuracy increases as the epoch number increases. The validation accuracy fluctuates around 0.87 when the epoch number reaches 10 and above while the training accuracy keeps increasing (Figure 3a).

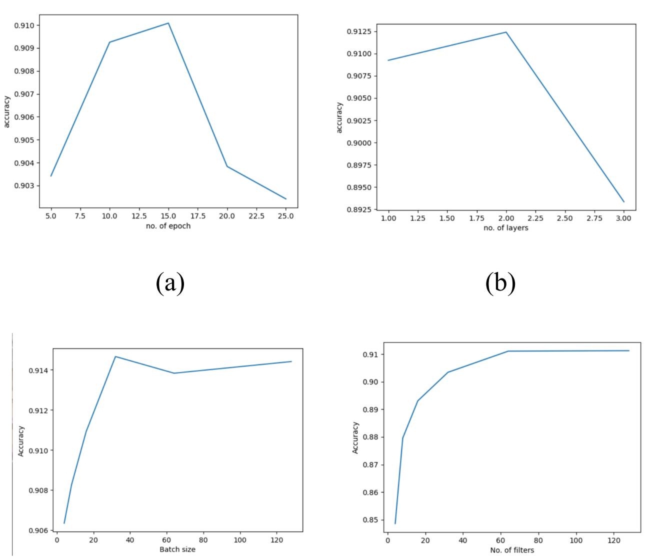
This indicates that the model was overfitting as we keep increasing epochs. There is no obvious improvement in validation accuracy as we increase the batch size (Figure 3b).

(a) (b)

Figure 3: Graph of accuracy of ANN model plotted against epoch value (a) and batch size (b)

4.4 Convolutional Neural Network

The learning process of the CNN model is largely affected by its hyperparameters. Particularly, four parameters: epochs, the number of convolutional layers, batch size of entries which traverse the network, and the number of feature filters employed in each convolutional layer were significant in ensuring optimality.



(a) (b)

(c) (d)

Figure 4: Graph of accuracy against epoch (a), no. of layers (b), batch size (c) and no. of filters (d) values of CNN

To gauge the most favourable values, multiple iterations were carried out by varying only the value for the interested parameter, keeping the other settings constant. Respective performance trends against the change in parameter value is represented in Figure 4.

In the observed trends, all parameters exhibited either a diminishing or goal-seeking performance trend as they exceeded a certain threshold. As the parameters increase in value, there is a high possibility that overfitting occurs, adding in excessive details into the model affecting prediction.

A direct approach to mitigate the overfitting effect will be to utilise the optimal parameter values. Moreover, performance is further improved by the inclusion of drop-out layers. By randomly dropping out neurons in the network layers, the likelihood of obtaining over-excessive details is largely reduced. After implementing these efforts, the CNN model achieved an impressive accuracy of 91.3%, surpassing the highest accuracy achieved by all the models in testing.

4.5 Model Comparison

An efficient model under our project setup will necessarily be required to prioritise the correct classification of classes. Hence, the accuracy matrix, which measures the proportion of correct guesses amongst all tries, was employed as our first choice discretion factor in quantifying algorithm performance. Taking into account that the test data of our interest may exhibit existing patterns, the Cohen Kappa Matrix, an inter-rater representation of accuracy, was also utilised to increase the generalisability of our analysis.

	DT	RF	ANN	CNN
Accuracy (%)	79.0	88.2	88.7	91.3
Cohen's Kappa (%)	76.5	86.9	87.4	90.4

Table 1: Model's Accuracy and Cohen's Kappa comparison

As seen in the above results, RF, ANN and CNN models are generally successful in the completion of the required task. Notably, CNN emerged as the frontrunner. This is attributed to CNN's proficiency in the use of image filters to extract spatially invariant features, which makes CNN exceptionally well-suited for the complex task of image classification.

Both RF and ANN models demonstrated similar commemorable accuracy and Cohen Kappa scores, exceeding expectations. The success in their performance is likely a result of the simplicity of our dataset, which possesses limited shape and colour features due to its small shape and grey scale nature. Such simplicity enables both RF and

ANN to overcome their inherent limitations in grasping image features through a spatial hierarchy, consequently, achieving performance levels that are nearly on par with the more sophisticated CNN.

Regrettably, the Decision Tree classification technique exhibited the poorest performance among all models. This outcome was largely anticipated, given that this model not only grapples with suboptimal performance in handling high-dimensional data but also struggles with embedded overfitting. The high dependence on the generation of decision conditions to train largely limits the algorithm's ability to learn in-direct patterns from the dataset, adversely affecting their generalisability in predicting image classes.

Other than basic functionality, time complexity also hold key in model evaluation. The decision tree algorithm is the most efficient in terms of time utilisation, as it only requires a single tree with conditions directly dependent on the amount of input data features. The ANN algorithm ranks succeeding, considering that its learning and prediction process durations are solely dependent on the number of layers present in the network. Conversely, the CNN and RF model proved to be the more intricate and time-inefficient models. In CNN, this is caused by the need for feature extraction using filters, causing factors such as image's height and width to be significant deciding elements contributing to high execution time. Whereas for the RF model, the presence of multiple trees nonetheless causes repetitive learning processes which adds additional computational load.

5. Conclusion

In conclusion, all four models are able to classify clothing images into distinct categories. Notably, CNN,ANN, and RT demonstrate commendable accuracy, with CNN slightly outperforming the others. However, when weighing accuracy against time complexity, ANN may be the more pragmatic choices for this project.

6. Reference

Breiman, L. (2001). *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/a:1010933404324

Charbuty, B., & Abdulazeez, A. (2021). Classification based on Decision Tree Algorithm for Machine Learning. *Journal of Applied Science and Technology Trends,2(01),*20-28. https://doi.org/10.38094/jastt20165

Cristea, P. D. (2009). Application of neural networks in image processing and visualization. *GeoSpatial Visual Analytics*, 59–71. https://doi.org/10.1007/978-90-481-2899-0_5

Goyal, C. (2023, February 27). *30 essential decision tree questions to ace your next interview (updated 2023)*. Analytics Vidhya. *https://www.analyticsvidhya.com/blog/2021/05/25-questions-to-test-your-skills-on-decision-trees/*

Maruyama, T., Hayashi, N., Sato, Y., Hyuga, S., Wakayama, Y., Watanabe, H., Ogura, A., & Ogura, T. (2018). Comparison of medical image classification accuracy among three machine learning methods. *Journal of X-Ray Science and Technology*, *26*(6), 885–893. https://doi.org/10.3233/xst-18386

Maximinusjoshus. (2021, June 8). *Understanding the concept of channels in an image*. Medium. https://medium.-com/featurepreneur/understanding-the-concept-of-channels-in-an-image-6d59d4dafaa9

Muses, M. (2019, August 22). *Shopping with your camera: How visual search is transforming eCommerce*. Medium. https://towardsdatascience.com/shopping-with-your-camera-how-visual-search-is-transforming-ecommerce-1bee5877994e

Quantitative Finance & Algo Trading Blog by QuantInsti. (2023, October 31). *Gini index: Decision tree, formula, calculator, Gini coefficient in machine learning*. Quantitative Finance & Algo Trading Blog by QuantInsti. https://blog.quantinsti.com/gini-index/

R, P. T. (2015). A comparative study on decision tree and Random Forest using r tool. *IJARCCE*, 196–199. https://doi.org/10.17148/ijarcce.2015.4142

Sharma, A. (2023, September 18). *4 simple ways to split a decision tree in Machine Learning (updated 2023)*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/06/4-ways-split-decision-tree/#:~:text=The%20-most%20widely%20used%20method,for%20the%20decision%20tree%20classifier.