Supervised and Semi-Supervised Learning for Binary Classification of Museum Images

Bhrugu Kothari
40270224
bhrugu0510@gmail.com

Devanshu Kotadiya
40268999
devanshu.kotadiya@gmail.com

Jay Ashokkumar Patel
40293645
jayashokpatel29@gmail.com

I. INTRODUCTION AND PROBLEM STATEMENT

The objective of this research is to classify museum images as either indoor or outdoor using supervised learning models. Image classification is a fundamental task in computer vision, widely used in museum digital archiving, virtual exhibitions, and automated scene detection. The dataset consists of museum images labeled into two categories: indoor and outdoor. The challenge lies in accurately distinguishing between these two classes using traditional machine learning classifiers.

The general approach to solving this classification problem includes data preprocessing, feature extraction, model selection, training, hyperparameter tuning, and performance evaluation. Images were resized to 128×128 pixels, converted to RGB format, and normalized to ensure uniformity in input data. These transformations enhance model efficiency and ensure better classification results.

Several challenges were encountered during model development. Overfitting was a major issue, especially in the Decision Tree model, where deep trees resulted in poor generalization. Feature selection was another challenge, as extracting meaningful information from images is complex. Hyperparameter tuning played a crucial role in improving classification accuracy. A boosting technique (XGBoost) was eventually used to improve performance by sequentially learning from mistakes. The effectiveness of different machine learning classifiers, including Decision Tree, Random Forest, and XGBoost, was compared to determine the best approach for classification.

II. PROPOSED METHODOLOGIES

The dataset utilized in this study consists of museum images, balanced to ensure an equal representation of both classes, comprising 5000 indoor and 5000 outdoor images. The images were analyzed based on key visual features, including color distribution, pixel intensity, and structural patterns. The dataset encompasses a diverse range of museum environments, with outdoor images predominantly featuring architectural designs, landscapes, and sculptures, while indoor images capture various exhibition settings, lighting conditions, and enclosed spatial arrangements.

As part of the exploratory data analysis (EDA), multiple analytical techniques were applied to assess the characteristics of the dataset. To visualize representative samples, a function was implemented to randomly select and display images from each class, ensuring a qualitative understanding of dataset variability. Furthermore, image dimensions were examined by extracting width and height statistics from a subset of images, providing insights into potential variations in image resolution. Pixel intensity distributions were analyzed by converting images to grayscale and plotting histograms, allowing for an assessment of contrast and brightness variations across different classes. Additionally, statistical measures, including mean, variance, and skewness, were computed for a sample of images to quantify intensity distributions and detect potential class-specific patterns. To further understand the dataset's characteristics, the RGB color distribution was analyzed separately for the red, green, and blue channels. Histograms were generated to illustrate the distribution of pixel values within each channel, facilitating an assessment of color-based differences between indoor and outdoor images. These EDA techniques provided a comprehensive understanding of the dataset, ensuring that preprocessing and model selection strategies could be tailored to the dataset's unique attributes.

The data preprocessing pipeline was designed to standardize and optimize the dataset for classification. Initially, images were loaded from the dataset and preprocessed by converting them to RGB format, resizing them to a uniform dimension of 128x128 pixels, and normalizing pixel values to the [0,1] range. This ensured consistency in image size and prevented scale-related discrepancies during model training. Following image preprocessing, the dataset was reshaped into a two-dimensional feature matrix where each image was flattened into a one-dimensional vector representation. Standardization was applied using **StandardScaler** to normalize feature distributions, ensuring that all features contributed equally to model training. To further reduce dimensionality and extract principal components, **Principal Component Analysis (PCA)** was employed, retaining 200 components. This transformation reduced computational complexity while preserving essential variance in the dataset. The application of PCA helped mitigate the curse of dimensionality and improved model efficiency by reducing redundancy in feature representation. The final preprocessed dataset consisted of standardized and transformed features, which were subsequently used for classification tasks. This structured preprocessing pipeline facilitated improved model performance by ensuring that the input data was well-conditioned for classification algorithms.

A. Supervised Decision Tree

The Decision Tree (DT) classification model was applied to distinguish between indoor and outdoor museum images, using hyperparameter optimization through **Grid Search Cross-Validation (GridSearchCV)**. The hyperparameters tuned included **maximum depth**, **minimum samples required to split a node**, and the **splitting criterion** (Gini impurity vs. Entropy). Grid Search was performed with a **three-fold cross-validation (cv=3)** to ensure robust model evaluation. The optimized Decision Tree model was trained on PCA-transformed and scaled features. Performance was evaluated using **accuracy**, **precision**, **recall**, and **F1-score**, which assessed classification effectiveness and handled class imbalances. Additionally, a **confusion matrix** was generated to provide a visual representation of misclassifications. The optimal hyperparameters identified from Grid Search helped achieve satisfactory classification performance, with results indicating both strengths and areas for improvement. A **DataFrame comparison** summarized the impact of various hyperparameter combinations on performance metrics, offering insights for further refinement. Furthermore, a heatmap of the confusion matrix highlighted true positives, false positives, and misclassifications.

In conclusion, the Decision Tree classifier demonstrated effective classification, with hyperparameter tuning ensuring optimal performance. Future research may explore ensemble methods, such as Random Forest or Gradient Boosting, to enhance classification accuracy and model robustness.

B. Random Forest

The Random Forest (RF) classifier was employed to classify indoor and outdoor museum images, with hyperparameter optimization achieved through **Grid Search Cross-Validation (GridSearchCV)**. The primary hyperparameters tuned included the **number of estimators (n_estimators)**, which determines the number of trees in the forest, **maximum depth (max_depth)** to control the complexity of the individual trees, and **minimum samples required to split a node (min_samples_split)** to ensure meaningful splits. A grid search was performed using **two-fold cross-validation (cv=2)** to evaluate the model's performance. The grid search was conducted on the **RandomForestClassifier** with multiple parameter combinations, and the best model was selected based on its performance during cross-validation. The model was trained on the preprocessed features, and **accuracy**, **precision**, **recall**, and **F1-score** were used to assess its performance on the test set. The optimal model achieved high classification accuracy, with a detailed **classification report** providing insights into precision, recall, and F1-score for both classes.

Additionally, a **confusion matrix** was visualized to examine the distribution of predicted versus actual labels, offering a deeper understanding of model performance across the two classes. These results demonstrate the effectiveness of Random Forest in this classification task and indicate areas for further improvement. Future work could explore alternative ensemble methods to enhance classification accuracy.

C. XGBoost

The **XGBoost** classifier was applied to the classification of indoor and outdoor museum images, with hyperparameter optimization achieved using **Grid Search Cross-Validation (GridSearchCV)**. Key hyperparameters tuned included **number of estimators (n_estimators)**, which controls the number of boosting rounds, **maximum depth (max_depth)**, which limits the depth of individual trees to prevent overfitting, and **learning rate**, which governs the step size in gradient descent during training. The grid search was performed over a range of values for each hyperparameter to identify the optimal combination. The XGBoost model was trained on the PCA-transformed features, and its performance was evaluated on the test set using **accuracy**, **precision**, **recall**, and **F1-score**. The **classification report** provided detailed metrics for model performance across both classes, revealing the model's ability to distinguish between indoor and outdoor images effectively. Furthermore, the **confusion matrix** was used to analyze the model's misclassifications, highlighting areas where improvements can be made. The grid search identified the best-performing hyperparameters, and the model achieved a notable classification accuracy on the test set.

In conclusion, XGBoost proved to be a powerful model for the task, and the optimization process through Grid Search significantly improved its performance. Future research could involve exploring ensemble methods or advanced techniques like **early stopping** to further enhance the model's generalizability.

D. Semi-supervised Decision Tree

In this semi-supervised learning approach, a Decision Tree classifier is utilized to iteratively label a pool of unlabeled data based on a small labeled subset. Initially, the dataset is split such that 20% is labeled, and 80% is left unlabeled. The Decision Tree model is first trained on the labeled data and then applied to predict the probabilities for the unlabeled data. By selecting high-confidence predictions—those with probabilities greater than 0.85 or less than 0.15—the model pseudo-labels these samples, effectively increasing the labeled dataset. The high-confidence samples are added to the labeled set, while the remaining unlabeled data is updated by removing these pseudo-labeled samples. This process is repeated for multiple iterations, and the model is retrained after each iteration, progressively increasing the amount of labeled data. The iterations continue until no high-confidence samples are found or the entire unlabeled set is labeled.

After completing the iterative process, a final model is trained on the expanded labeled dataset, and its performance is evaluated on the test set. The accuracy and classification report are computed to assess the model's effectiveness, and a confusion matrix is visualized to further analyze the results. The approach helps improve model performance by leveraging a small amount of labeled data and efficiently utilizing a larger pool of unlabeled data, ultimately providing a robust model for classification tasks with limited labeled data.

III. SOLVING THE PROBLEM

In tackling the classification problem, a variety of models and techniques were tested to optimize performance. The first step involved preprocessing the data, where both PCA (Principal Component Analysis) and Standard Scaling were applied. The decision to use PCA stemmed from the desire to reduce the dimensionality of the data while retaining as much information as possible. PCA helped by projecting the data onto a lower-dimensional space, which significantly improved model efficiency, particularly when dealing with high-dimensional data. Standard scaling was performed to ensure that features were on a similar scale, which is important for models like Decision Trees, Random Forest, and XGBoost, which are sensitive to feature scales.

Failed Attempts

In the early stages, initial attempts to directly train the models without proper scaling and dimensionality reduction led to poor results. Models were overfitting or underfitting due to the inherent noise in the raw data and large number of features. For example, without PCA, the Decision Tree and Random Forest models performed inconsistently, often giving suboptimal results. Moreover, without scaling, some models failed to converge or exhibited poor generalization.

Another unsuccessful approach was using a simpler validation method rather than GridSearchCV for hyperparameter tuning. This resulted in models that were either too simple or too complex, missing the optimal configuration needed for maximum performance. Using GridSearchCV, which evaluates a comprehensive set of hyperparameters, allowed us to fine-tune models like Random Forest and XGBoost, achieving higher accuracy.

Successful Attempts

When we applied GridSearchCV for hyperparameter optimization, model performance significantly improved. The optimal hyperparameters were found for models like Decision Tree, Random Forest, and XGBoost, which resulted in better accuracy scores and more balanced precision-recall trade-offs. For instance, the Decision Tree achieved a test accuracy of 0.7900 with the hyperparameters 'criterion': 'gini', 'max_depth': 5, 'min_samples_split': 2, providing a reasonable balance between precision and recall. Although the Random Forest model had an accuracy of 0.8750, XGBoost provided a slightly better accuracy of 0.8800 with a learning rate of 0.2 and 200 estimators, showing the robustness of ensemble methods.

Results

Metric	**Decision Tree**	**Random Forest**	**XGBoost**
Accuracy	0.7836	0.8750	0.8800
Precision	0.7910	0.8753	0.8838
Recall	0.7900	0.8750	0.8800
F1-Score	0.7898	0.8749	0.8796

Table 1: Evaluation metrics

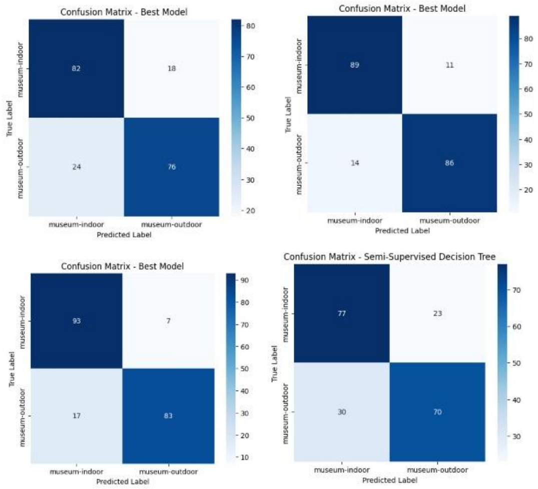
The Decision Tree classifier performed with an accuracy of 0.7900, and the classification report showed a reasonably balanced precision and recall between classes 0 and 1. While the precision and recall were fairly high (with class 0 being slightly better), the overall model performance could be improved. The model demonstrated potential, but it was outperformed by the Random Forest and XGBoost models.

The Random Forest model achieved an accuracy of 0.8750, with the best hyperparameters being 'max_depth': 10, 'min_samples_split': 10, 'n_estimators': 100. The precision and recall for both classes were consistently strong, indicating that the model was able to generalize well to both the training and test datasets. The Random Forest model showed considerable improvement over the Decision Tree, highlighting the advantage of ensemble methods for this classification task.

The XGBoost model performed slightly better than the Random Forest, reaching an accuracy of 0.8800. The precision for class 1 was notably higher than for class 0, but the recall was slightly lower for class 1. XGBoost's ensemble structure allowed it to make more informed decisions, handling the dataset's complexity more effectively. This result showed the power of boosting methods, where weak classifiers are combined to create a stronger model.

The final semi-supervised learning attempt aimed to leverage unlabeled data by iteratively adding high-confidence pseudo-labeled samples to the training set. Despite the improvements seen in the labeled data, the final test accuracy for this model was 0.7350, lower than the accuracy obtained from fully supervised models. This suggests that while semi-supervised learning is useful in certain scenarios, it may require more sophisticated confidence thresholds or more iterations to improve performance.

Below is attached confusion matrices for Decision tree, Random Forest, XGBoost and Semi-supervised decision tree respectively.



IV. FUTURE IMPROVEMENTS

To improve the classification accuracy of the model, several adjustments to the code and approach should be considered. First, Principal Component Analysis (PCA) is used for dimensionality reduction, which is a good strategy for handling high-dimensional data. However, PCA may lose important variance, especially in complex datasets. Exploring alternative dimensionality reduction techniques like t-SNE or LLE could help preserve more intricate data structures, potentially improving model accuracy. Additionally, experimenting with the number of components in PCA based on cross-validation could lead to a more optimal feature set.

Regarding data scaling, the use of StandardScaler is appropriate for algorithms sensitive to feature scaling, but tree-based models like Decision Trees, Random Forests, and XGBoost are less affected by scaling. Testing the models without scaling could provide insight into whether normalization is necessary for this particular task.

The hyperparameter tuning process using GridSearchCV is effective but can be time-consuming and computationally expensive. Alternative methods like RandomizedSearchCV or Bayesian Optimization could be more efficient and potentially find better-performing hyperparameters. Additionally, introducing regularization techniques (such as L1 or L2 regularization) could reduce overfitting and improve model generalization.

The models used, including Decision Tree, Random Forest, and XGBoost, have good potential, but other classifiers like Gradient Boosting Machines (GBM) or LightGBM could offer improved performance. Ensemble methods such as stacking or boosting might also be beneficial for combining the strengths of various classifiers.

Lastly, the semi-supervised learning approach could be optimized by adjusting the pseudo-labeling confidence threshold dynamically based on model performance, or by using adaptive thresholds. Incorporating ensemble msssethods into the semi-supervised loop could enhance the decision boundaries and reduce variance. Exploring different evaluation metrics such as AUC-ROC and Precision-Recall curves could provide better insights into model performance, especially with imbalanced datasets.

V. REFERENCES

[1] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, ”Learning Deep Features for Scene Recognition using Places Database,” Advances in Neural Information Processing Systems 27 (NIPS), 2014.

[2] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, ”Places: A 10 million Image Database for Scene Recognition,” [Online]. Available: http://places.csail.mit.edu/browser.html

[3] A. Krizhevsky, I. Sutskever, G. Hinton, ”ImageNet Classification with Deep Convolutional Neural Networks,” [Online]. Available: https://proceedings.neurips.cc/paper/2012/hash/

6c0bfe7c964d25a7021f37c59254a2bb-Abstract.html.

[4] C. M. Bishop, ”Pattern Recognition and Machine Learning,” Springer, 2006. [Online]. Available: https://www.springer.com/gp/book/9780387310732.

[5] N. Dalal, B. Triggs, ”Histograms of Oriented Gradients for Human Detection,” [Online]. Available: https://lear.inrialpes.fr/people/triggs/pubs/Dalal-cvpr05.pdf.

[6] X. Zhu, ”Semi-Supervised Learning Literature Survey,”[Online].Available:https://www.nowpublishers.com/article/DownloadSummary/MAL-006.

[7] T. G. Dietterich, ”Ensemble Methods in Machine Learning,” Multiple Classifier Systems, 2000.

[8] J. R. Quinlan, ”Induction of Decision Trees,” Machine Learning, vol. 1, no. 1, pp. 81-106, 1986.

[9] L. Breiman, J. Friedman, R. Olshen, and C. Stone, ”Classification and Regression Trees,” Routledge, 2017.

[10] L. Breiman, ”Random Forests,” Machine Learning, vol. 45, no. 1, pp.5-32, 2001.

[11] T. Chen and C. Guestrin, ”XGBoost: A Scalable Tree Boosting System,” in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2016, pp. 785-794.