Brand Logo Detection:

Unmasking the counterfeits

Contents

Abstract

Our research's main goal was to develop a machine learning model for detecting fake logos that is accurate at classifying authentic logos from fake ones across a range of businesses. We aimed to safeguard consumer interests, maintain market integrity, and uphold brand reputation through achieving this goal. We gathered a significant number of logos from various sources, including authentic and fake samples. Modern deep learning architectures were used to train the model on this dataset, and CNN was used to increase detection precision. With a precision of 91% and a recall rate of 98% on the original images, and 87% and 55% on fake images, respectively, the model has demonstrated a high level of detection accuracy. The results of our experiment highlight the importance and practicality of Utilizing modern tools to stop the spread of fake logos. By accurately recognizing and resolving this problem, we hope to make the market a safer place, increase consumer confidence, and advance the integrity of brands.

Introduction

The problem of fake brand logos, also known as counterfeit or knockoff logos, poses a significant threat to various industries and consumers worldwide. These logos are designed to imitate well-known brands but are unauthorized and often associated with selling fake or inferior products. This practice undermines brand reputation and compromises consumer safety. Our study is focused on identifying and detecting fake logos using advanced machine learning techniques.

The negative effects of fake logos on customers and businesses are what inspired this initiative. Fake logos mislead customers into purchasing poorer or even unsafe products, breaking their faith in businesses, and lowering their level of happiness. The existence of fake logos also compromises legitimate businesses that invest in brand equity and quality requirements.

To defend consumer interests, trustworthy brand reputation, and market integrity, an effective fake logo identification system must be created. Our objective is to develop a machine-learning model for a system that can enable customers, companies, and regulatory bodies to recognize and stop the intentional use of counterfeit logos.

Our research has shown that false logos are widely used across businesses, highlighting how important it is to solve the problem. To stop this growth, effective methods of detection are required. Additionally, we've developed innovative techniques that can help detect real and fake logos, providing better defense against false advertising.

Dataset

The dataset consists of 12000 images of 100 brands with 100 original images and 20 fake images of each brand. We analysed different datasets from the previous research. In those datasets, there are only original logo images. Whereas for this project, there is a need for both fake and original images. We didn’t find any proper dataset according to the requirements. So, the custom dataset was created.

The images were collected from various sources. The original logos were collected from each brand's official site and Google. While in the case of fake logos, for some brands' they were readily available on Google and Pinterest and were downloaded directly from these websites. For brands whose fake logos were not available, they were generated using the "Brand Crowd" website and PowerPoint.

All the downloaded images were in different formats, for example, some images were in JPEG while some were in PNG format. So, by using the PIP library from Python, the format of every image has been converted to JPG. After this, all the images were renamed to some unique format.

There are “Four” categories in the dataset namely, “Transportation, Food and Beverages, Fashion, and Electronics” where each category has separate folders as “fake” and “original” under the “brand's” folder in google drive.

Method of Analysis

In the process of the analysis, ETL (Extract Transform & Load) and modelling are the main steps of the process. The preprocessing of the data is carried out in the ETL process. In the modeling process, machine learning models are developed to analyze data, make predictions, and gain insights by learning patterns and relationships within the data.

The most common machine learning models for the classification of the images are Convolutional Neural Network (CNN) model and Support Vector Machine (SVM) model. While working with these models, we faced some issues like crashing the notebook.

When we were working with the SVM model, we tried different values of the regularization parameter (C) to get good accuracy for the model. The primary goals of the models are to achieve the highest accuracy, high precision, and high recall in predicting outcomes or classifying data.

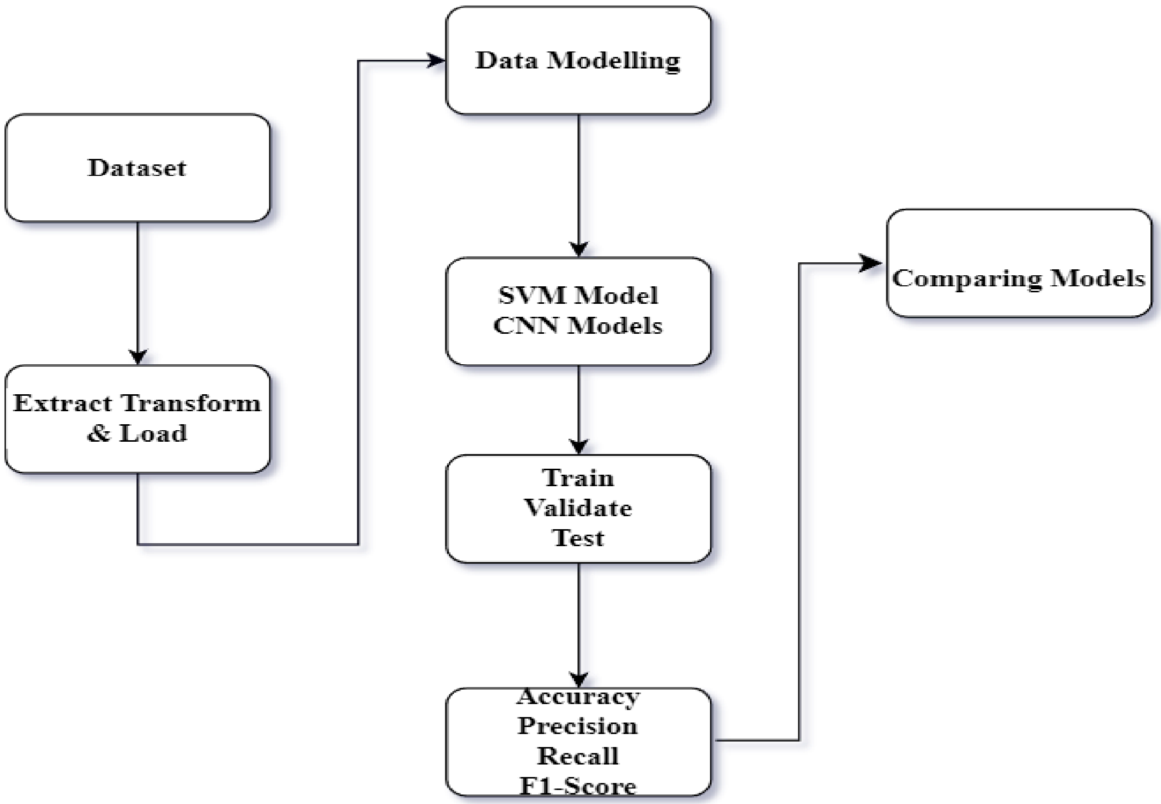
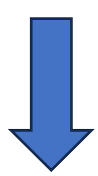


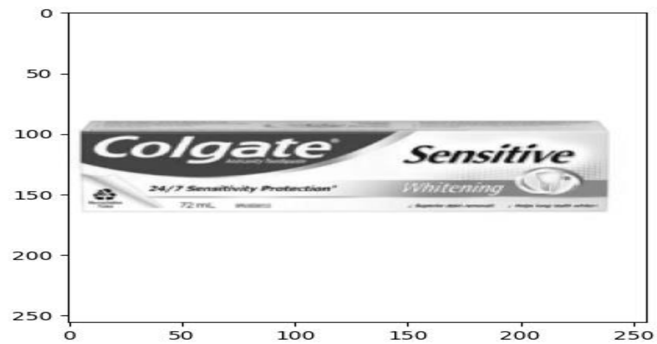
Fig (4.1) - Flow chart of fake brand logo detection

ETL

In the ETL process, the images are converted to grayscale to reduce the dimensions and extract the features. After that images are resized to a fixed size of 256x256 pixels.

Original Image





Grayscaled and resized image

Modeling

Model 1 – SVM

SVM (Support Vector Machine) is a supervised machine learning algorithm used for classification and regression tasks. SVM models have the ability to handle complex patterns, capture non-linear decision boundaries, and handle high-dimensional data efficiently. The dataset we are using has two clearly separated classes original and fake. SVM can be used to classify images based on their features. The model's accuracy is used as the primary metric to evaluate its performance on the validation and test datasets.

Model 2 – Baseline Model (CNN)

A convolutional neural network (CNN) is a type of deep learning. CNN models have the ability to learn complex features and optimal representations and classification functions directly from the raw data to the final data. It is good for capturing spatial relationships, extracting meaningful features, and providing robustness to variations in the position and appearance of objects within images.

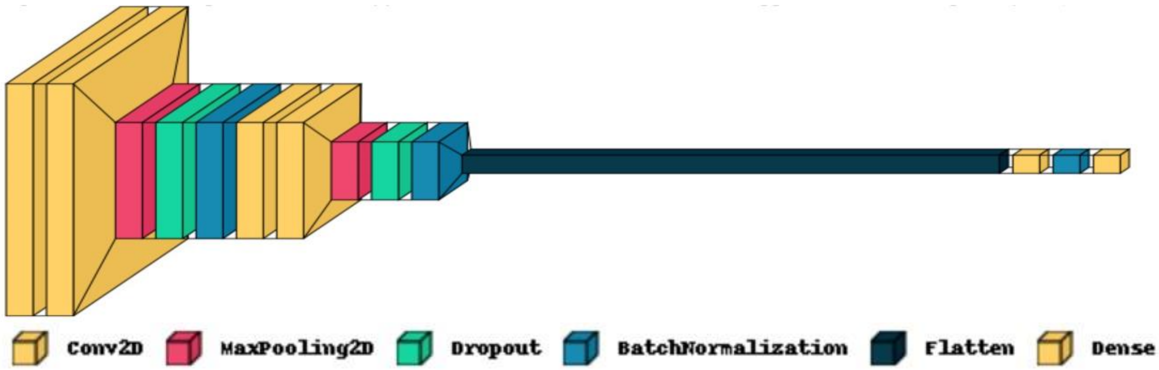


Fig (4.2) – Baseline Model

In this model, CNN classified the fake and original logos in the data. The model follows a common CNN design pattern, and it consists of 14 layers. There were four 2D Convolutional layers with 128 filters each having a size of (3,3), the activation function as 'relu', and the 'same' padding strategies used in the layers. Two max-pooling layers of pool size (2,2) are used in this model to reduce the spatial dimension. The model includes three batch normalization and two dropout layers with a dropout rate of 0.2. A flatten layer is used in the model for converting 2D to 1D features. Two fully connected layers are included in the model, one as the output layer with the activation function as 'softmax'.

CNN Model -02

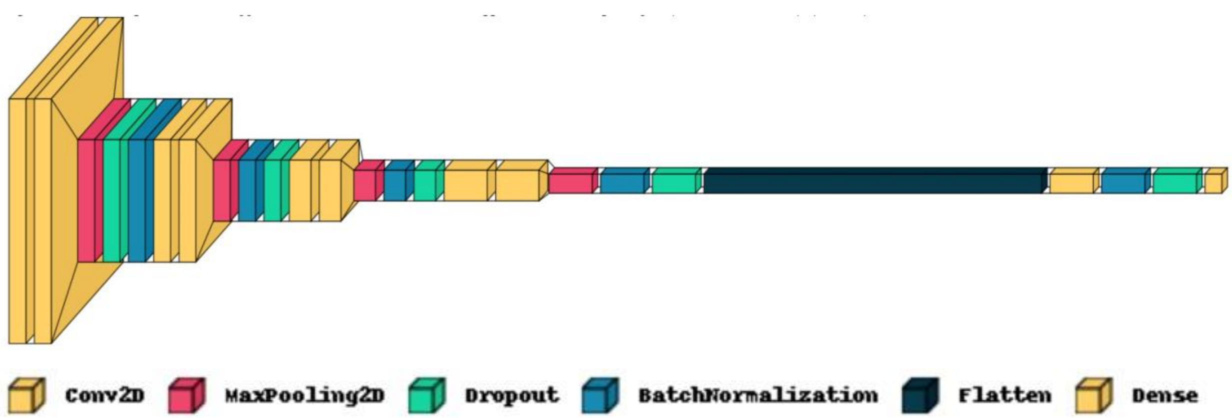


Fig (4.3) – CNN Model -02

We made configurations in the baseline model because of the overfitting of the model. For countering the issue, more layers are added to the model, the number of filters was changed in the convolutional layers and the dropout rate in the dropout layer also changed. For the final model, we made a configuration in the previous model. This model has 25 trainable layers. There were eight 2D Convolutional layers in this model, the activation function as ‘relu’, and the ‘same’ padding strategies used in the layers. To reduce the spatial dimensions in the model, four max-pooling layers of pool size of (2,2) are used. The model has five dropout layers with dropout rates of 0.3,0.4,05,0.5 and 0.5 respectively. A flattened layer is used in the model for converting 2D to 1D features. Five batch normalization layers are also in it. The model contains two fully connected layers, one of which is the output layer and has the activation function softmax. The early stopping technique is used for training the model to prevent overfitting and the optimal point of convergence.

Final CNN Model

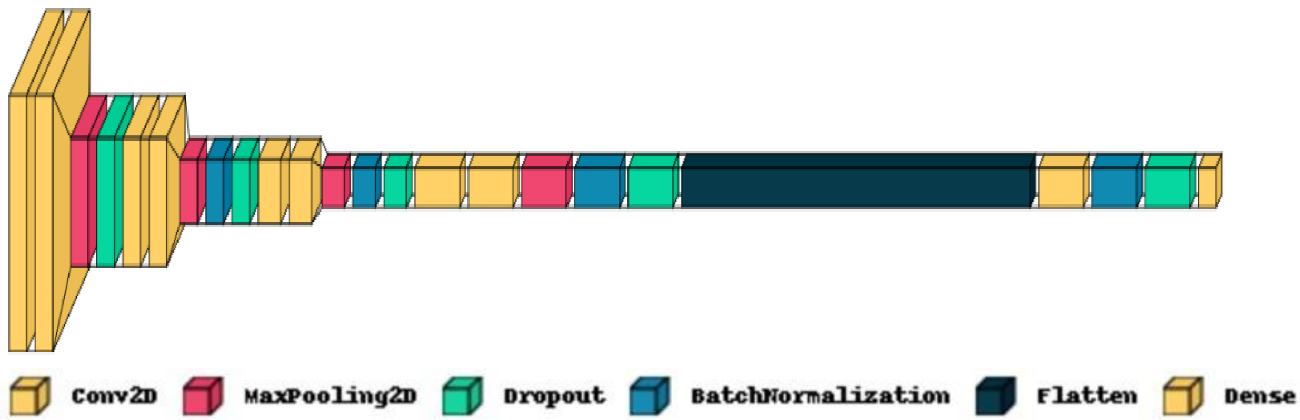


Fig (4.4) – Final CNN Model

The CNN model -02 had low precision and recall scores for the fake class, so we made some configurations in the model architecture. The final CNN model has 24 trainable layers. There are eight 2D Convolutional layers in this model with the 'relu' activation function and 'same' padding strategy used in each layer. To reduce the spatial dimensions in the model, we applied four max-pooling layers with a pool size of (2, 2). To prevent overfitting, we added five dropout layers with a dropout rate of 0.5. Additionally, four batch normalization layers are used to improve the stability and convergence of the model. The model also includes a flatten layer to convert the 2D features to 1D before passing them to the fully connected layers. The model contains two fully connected layers, one of which is the output layer and has the activation function ‘softmax’.

Results

Accuracy

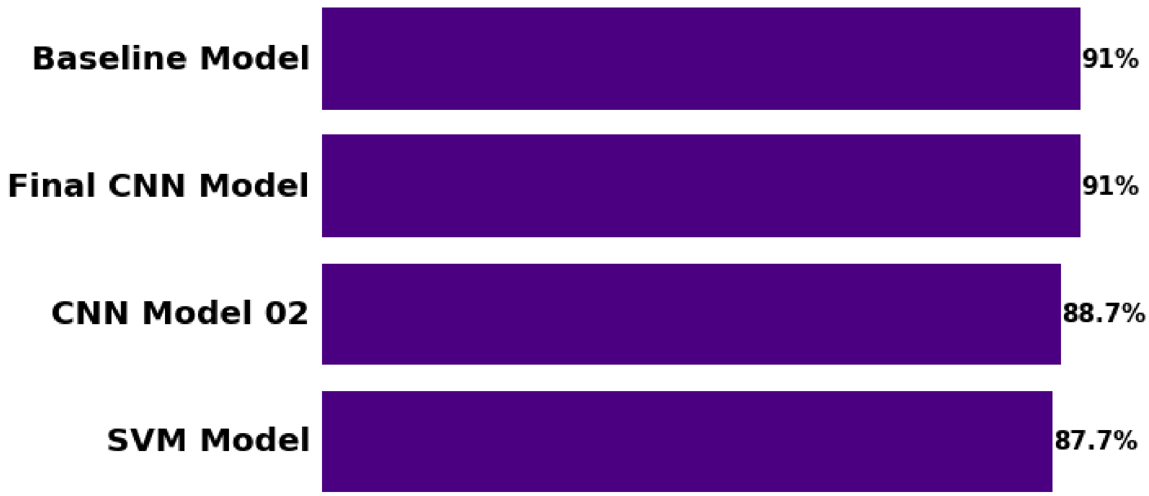


Fig (5.1) - Models Accuracies

Precision

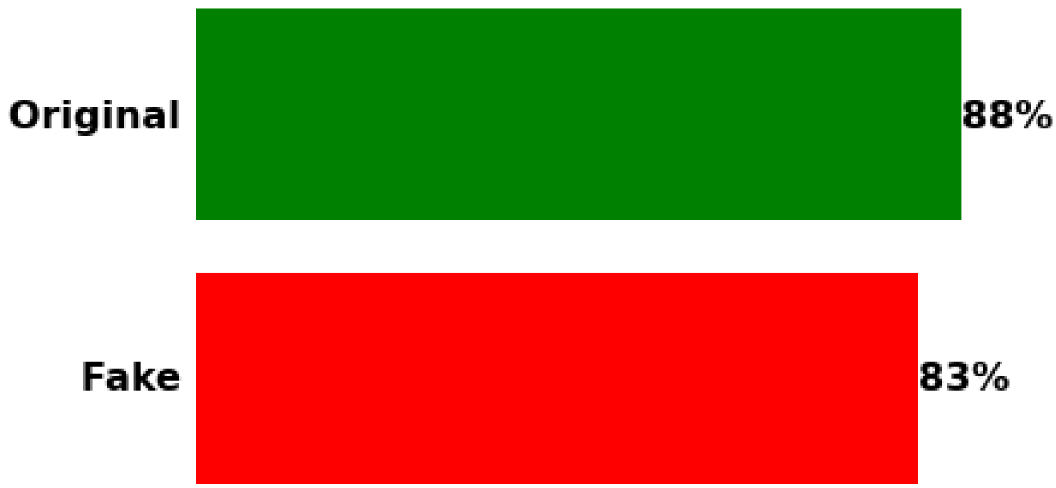


Fig (5.2) – Precision of CNN Model-02

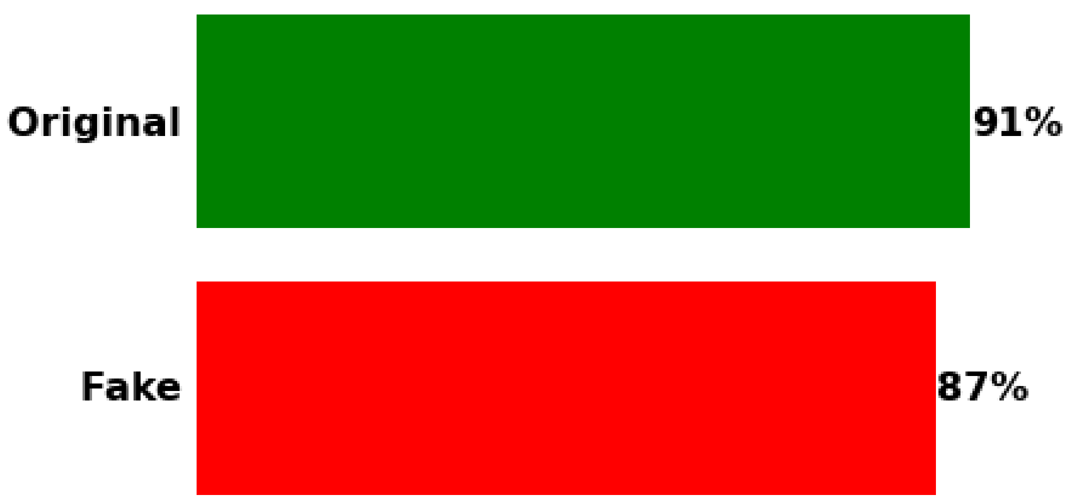


Fig (5.3) - Precision of Final CNN Model

Recall

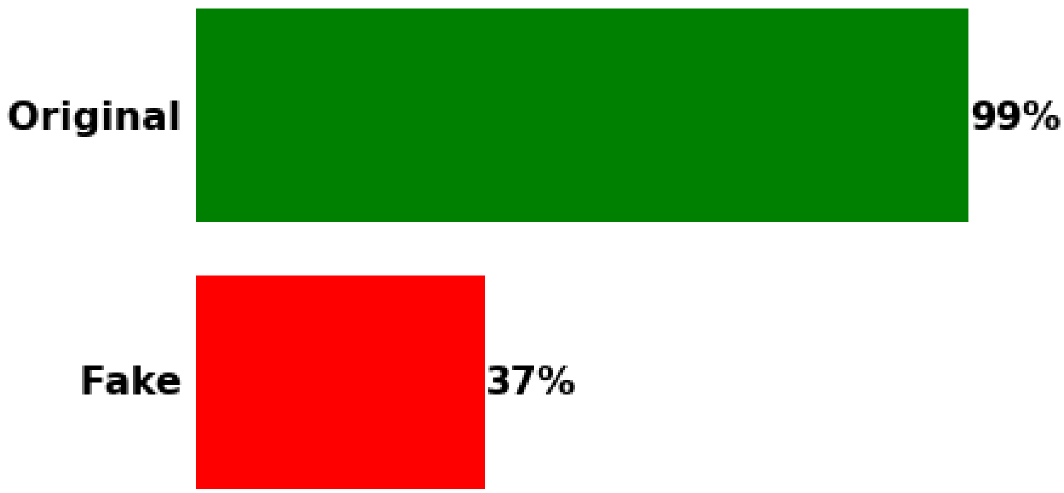


Fig (5.4) – Recall of CNN Model-02

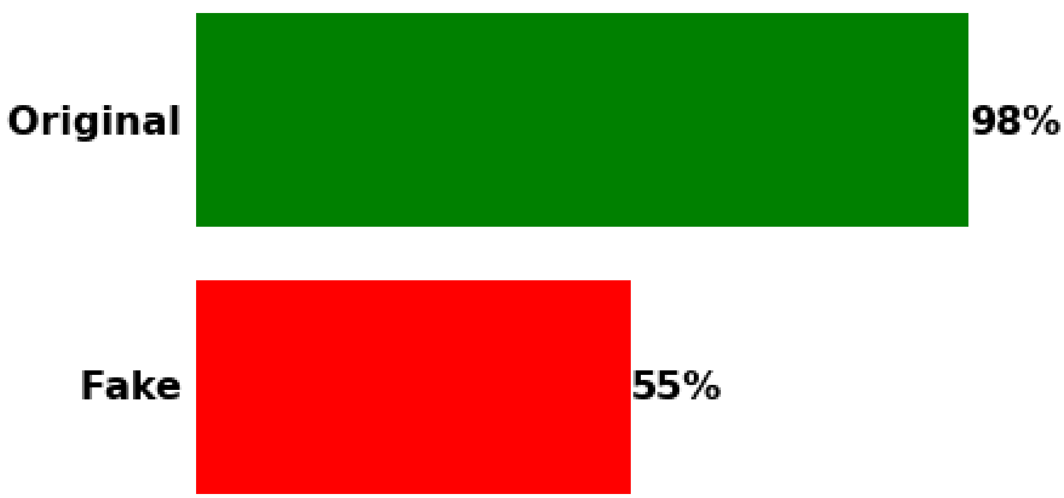


Fig (5.5) – Recall of Final CNN Model

Discussion

In the project, four models are used, and they are evaluated using accuracy, precision, and recall.

In terms of accuracy, the baseline model and the Final CNN model have higher accuracy than the other two models. Both models achieved an accuracy of 91%, indicating that models correctly predicted 91% of all instances in the data. The CNN Model-02 and the SVM model were able to correctly classify 88.7% and 87.7% of the data, respectively.

The metric accuracy might not be sufficient in our project because the number of original logos is more than fake logos. Due to the differences in the number of logos, the accuracy doesn’t provide the model performance on both classes. It more focuses on the majority classes than the minority classes. Precision and recall provide valuable insights into the classification model and evaluate the effectiveness in identifying fake and original logos.

While evaluating precision, the final model outperformed CNN Model-02. The final model demonstrated superior predictive capability for both the "original" and "fake" classes, correctly identifying 91% and 87% of the instances for each class, respectively. In comparison, CNN Model-02 achieved a precision rate of about 88% for the "original" class and "fake" class, respectively.

When evaluating recall, both the Final CNN model and CNN Model-02 performed well for the original class but struggled with the fake class. However, the Final CNN model showed better recall scores compared to CNN Model-02. The Final CNN model correctly predicted 98% of the original images and 55% of the fake images from the dataset. The correct prediction of the CNN Model -02 was about 99% for the original class and 37% for the fake class from the data.

Limitations

The project has some limitations. One of these limitations is that the model used in the project is not performing well in predicting the fake class. Another limitation lies in the imbalanced data, with a predominance of original images in the dataset, leading to poor performance in predicting the fake class.

Conclusion

The model has shown a high accuracy rate of 91%, which is a clear indication of how well it can distinguish between Original and Fake logos. The model also shows great potential for use in various real-world applications.

Future Work

The project has potential for future improvements to enhance its overall performance and effectiveness. One of the tasks for the future is to create a well-balanced dataset (equal number of images for both classes). The exploration of different architectures and techniques would improve the overall model capabilities and performance. The improvement of precision and recall score of the fake class would be helpful to make correct predictions in the fake class. Deployment of the model is also a prominent aspect for future improvement in the project.

Acknowledgments

We would like to express our gratitude to the following individuals and organizations for their invaluable contributions to this project:

Prof. Umair Durrani : Model optimization

Prof. Manjari Maheshwari: Development of base model

Prof. Pratik Bedi: Selection of topic.

We are also thankful to our friends for their continuous support and encouragement throughout this endeavor.

Lastly, we want to acknowledge the support provided by St Clair College in facilitating the resources and environment necessary for the successful completion of this project.

Thank you all for being a part of this journey.

Akshay Shaji

Deelan Joston Monterio

Delphina Dominic Carval

Sai Kumar Yamsani

References

1.4. Support Vector Machines. (n.d.). Scikit-Learn. Retrieved July 26, 2023, from https://scikit-learn/stable/modules/svm.html

Bianco, S., Buzzelli, M., Mazzini, D., & Schettini, R. (2017). Deep Learning for Logo Recognition. *Neurocomputing*. https://doi.org/10.1016/j.neucom.2017.03.051

Convolutional Neural Network (CNN) | TensorFlow Core. (n.d.). TensorFlow. Retrieved July 28, 2023, https://www.tensorflow.org/tutorials/images/cnn

Gupta, M., Singhal, S., Nyamati, V., & Ramasamy, S. (2021). *Logo Infringement Detection using Machine Learning*.

Precision-Recall—Scikit-learn 1.3.0 documentation. (n.d.). Retrieved July 26, 2023, from https://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html

Team, K. (n.d.). *Keras documentation: Conv2D layer*. Retrieved July 28, 2023, from https://keras.io/api/layers/convolution_layers/convolution2d/