Efficient Dataset Distillation for Cat and Dog Image Classification

Batuhan Sal
150210316
sal21@itu.edu.tr

Ömer Erdağ
150210332
erdag21@itu.edu.tr

Serdar Biçici
150210331
bicici21@itu.edu.tr

***Abstract*—Dataset distillation helps make image datasets smaller while keeping the important information needed for training accurate models. This project looks at how to directly distill the LSUN cat and dog dataset, with the idea that smaller datasets can still keep the accuracy of larger ones. We use three methods: random sampling, clustering, and data augmentation. First, we train convolutional neural networks on the full dataset to see the baseline accuracy. Then, we use the three distillation methods to create smaller datasets and retrain the models. We compare the accuracy and efficiency of these smaller datasets to the original. We expect that the smaller datasets will be almost as accurate as the full dataset but will take up less space and require less processing time. This study will help us understand how these distillation methods work to keep the important parts of images for correctly identifying cats and dogs.**

***Index Terms*—Distillation, cat & dog, classification, random, clustering, selective clustering, augmentation, CNN**

NOTEBOOK

For the application of this project, you can view the notebook from in Kaggle.

TEAM

Batuhan Sal - Constructing base model, Data Augmentation
Ömer Erdağ - Constructing base model, Random Distillation
Serdar Biçici - Data collection, Clustering Distillation

I. PROBLEM

The LSUN cat and dog dataset has millions of images, making it very large and hard to work with. Training models on such a big dataset needs a lot of computing power, storage, and time. This project aims to solve this problem by making the dataset smaller without losing important information needed for accurate image classification.

Our approach involves three methods to make the dataset smaller: random sampling, clustering, and data augmentation. First, we will train a model on the full dataset to see how accurate it can be. Then, we will use each of the three methods to create smaller versions of the dataset and retrain the models on these smaller datasets.

The main problem we are tackling is to see if these smaller datasets can keep the accuracy close to the full dataset while being much smaller and easier to work with. We will compare how each method affects the model's accuracy and efficiency. Our goal is to find out which method works best for keeping the important features needed to correctly classify cat and dog images.

II. HYPOTHESIS

We hypothesize that direct dataset distillation can effectively compress the LSUN cat and dog dataset to just 10% of its original size while maintaining comparable accuracy levels for training image classification models. By applying three specific distillation techniques—random sampling, clustering, and data augmentation—we believe we can create smaller, yet informative, datasets that enable efficient training of convolutional neural networks. We expect that models trained on these distilled datasets will achieve similar results to those trained on the full dataset, demonstrating that it is possible to significantly reduce dataset size without sacrificing classification performance.

III. LITERATURE SURVEY

Deep learning has seen significant advancements in fields like computer vision and natural language processing due to its powerful feature extraction capabilities. However, training on large datasets can be costly and time-consuming. Dataset distillation offers a solution by compressing large datasets into smaller, synthetic ones, reducing memory and compute requirements while maintaining model performance.

One study on optimizing convolutional neural network parameters for classifying 8,000 cat and dog images achieved an accuracy of 88.31% by using a CNN for feature learning followed by an artificial neural network (ANN) binary classifier. This highlights the potential for high accuracy even with smaller datasets.[1]

Practical applications, like Samuel Cortinhas' MNIST dataset distillation project on Kaggle, demonstrate the effectiveness of this approach. His work shows that smaller, distilled datasets can maintain performance while being more efficient to use.[2]

Building on these insights, our project aims to apply and compare different distillation techniques—random sampling, clustering, and data augmentation—to reduce the LSUN cat and dog dataset size while preserving classification accuracy.

IV. Methods

A. Base Model

The base model is a convolutional neural network designed for image classification tasks. It consists of several layers that perform operations like convolution, batch normalization, activation, max pooling, flattening, and dense layers with dropout for regularization. Here's a breakdown of the key components:

1) **Convolutional Layers**: The network starts with a series of convolutional layers, each followed by batch normalization and rectified linear unit (ReLU) activation function. These layers extract features from the input images by applying a set of learnable filters to capture patterns at different spatial scales.
2) **Max Pooling**: After each convolutional block, max pooling layers downsample the feature maps to reduce spatial dimensions while retaining the most important information. This helps in reducing computational complexity and controlling overfitting.
3) **Flattening**: The flattened layer converts the 2D feature maps into a 1D vector, which serves as the input to the fully connected layers.
4) **Fully Connected Layers**: The flattened features are passed through fully connected dense layers with batch normalization and ReLU activation. These layers capture high-level patterns and relationships in the feature space.
5) **Dropout**: Dropout layers are added to prevent overfitting by randomly deactivating a fraction of neurons during training.
6) **Output Layer**: The final dense layer with a sigmoid activation function outputs the predicted probability of the input image belonging to a particular class, binary classification in this case.

The model is compiled using the Adam optimizer and binary cross-entropy loss function. During training, the network learns to minimize the loss function by adjusting the weights and biases of its layers using backpropagation.

```
# Base Model
Test accuracy: 0.9120000004768372
```

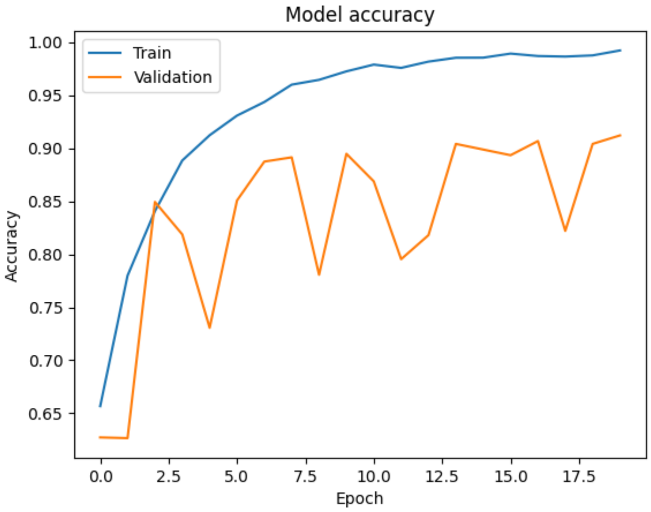

Fig. 1. Base Model Accuracy

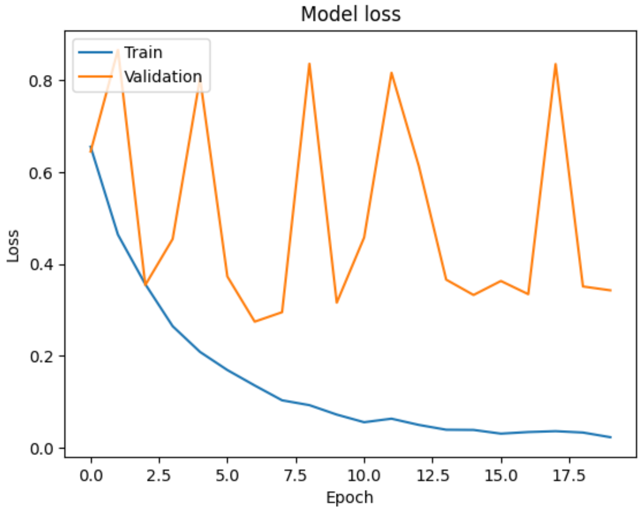

Fig. 2. Base Model Loss

B. Random Distillation

Random distillation is a technique used to reduce the size of a dataset by randomly sampling a fraction of the original data while ensuring that the class distribution remains balanced. The process involves the following steps:

1) **Dataset Splitting**: The original dataset is divided into training and test sets. The training set is further divided into smaller subsets by randomly sampling a certain percentage of data points. In this approach, we sample 50%, 25%, and 10% of the original training dataset.
2) **Stratified Sampling**: To maintain the class distribution, stratified sampling is employed. This ensures that each class is represented proportionally in the sampled subsets. For example, if the original dataset contains equal numbers of cat and dog images, the sampled subsets will also contain an equal proportion of cat and dog images.
3) **Model Training**: A convolutional neural network model is trained on each sampled subset independently. The purpose of training on smaller subsets is to create models that are computationally less intensive while still capturing the essential features required for accurate classification.
4) **Model Evaluation**: The trained models are evaluated on the test dataset to assess their performance in terms of classification accuracy and loss. This step helps determine the effectiveness of the distillation process in reducing the dataset size without significantly compromising model performance.
5) **Model Saving and Analysis**: The trained models with the best performance metrics are saved for future use. Additionally, the training and validation accuracy curves may be plotted to analyze the learning behavior of the models over epochs.

By applying random distillation, we aim to create smaller, yet representative, subsets of the original dataset that can be used to train efficient convolutional neural network models for image classification tasks. This approach enables us to reduce computational resources and training time while maintaining satisfactory classification performance.

Their test accuracies are:

```
# 50%
Test accuracy: 0.7480000257492065
Test loss: 1.298595666885376

# 25%
Test accuracy: 0.8133333325386047
Test loss: 0.6526797413825989

# 10%
Test accuracy: 0.7553333044052124
Test loss: 0.9564400315284729
```

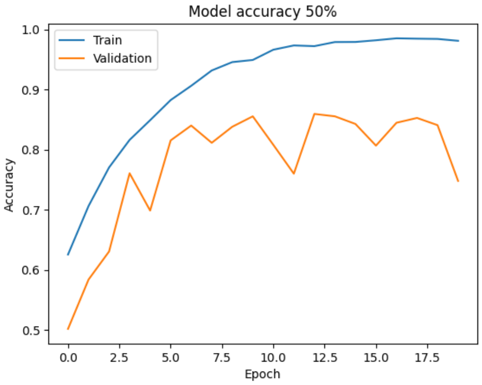

Fig. 3. Random Distillation (50%)

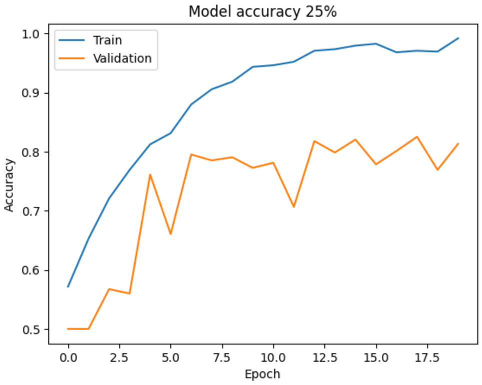

Fig. 4. Random Distillation (25%)

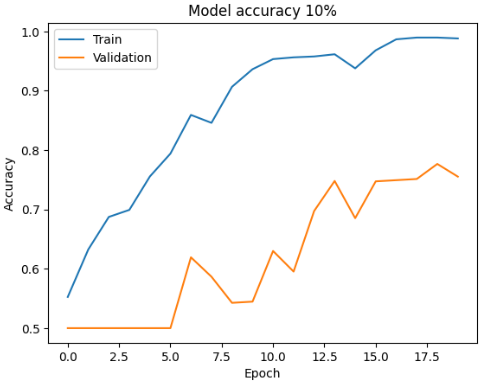

Fig. 5. Random Distillation (10%)

As can be seen, there is no determinant result of random distillation. However, if the cost is wanted to be reduced without any expectation, this technique is useful.

C. *Clustering Distillation*

Image clustering is a technique used to group similar images together based on their visual features. The `ImageClusterSampler` class provides functionalities for clustering images and selecting diverse samples from each cluster. Here is the theoretical approach for the clustering method:

1) **Initialization**: Initialize the `ImageClusterSampler` with the image data, corresponding labels and the desired number of clusters.
2) **Clustering**: Implement the clustering algorithm, which involves two main steps:
   - **Feature Extraction**: Convert each image in the dataset into a feature vector using techniques like flattening or dimensionality reduction.
   - **Clustering Algorithm**: Apply a clustering algorithm (PCA) to group the feature vectors into clusters based on their similarity.
3) **Visualization**: Visualize the clusters in a scatter plot by projecting the feature vectors onto a 2D space. Each cluster is represented by a different color and the scatter plot provides insights into the distribution of images in the feature space.
4) **Selection of Diverse Samples**: Implement a method to select a specified number of diverse samples from each cluster. This involves partitioning the feature space into a grid and randomly selecting samples from each grid cell. The goal is to ensure that the selected samples are evenly distributed across the feature space, capturing the diversity of images within each cluster.
5) **Visualization of Selected Samples**: Plot the selected samples on the scatter plot to visualize their distribution within the clusters. This helps assess the effectiveness of the sampling strategy in capturing diverse image representations.

By following these steps, the `ImageClusterSampler` class facilitates the clustering of images and the selection of diverse samples, which can be useful for tasks like dataset distillation and active learning.

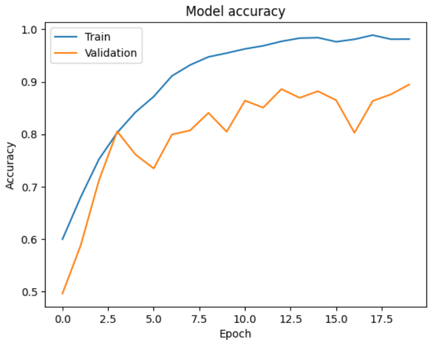

Fig. 6. Cluster Distillation

```
# Clustering Distillation
Test accuracy: 0.8946666717529297
Test loss: 0.4023839235305786
```

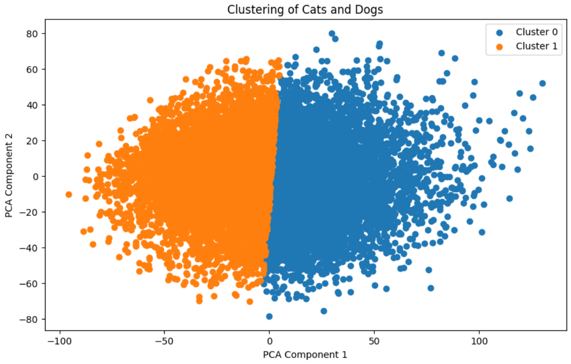

Fig. 7. Cat and Dog Cluster

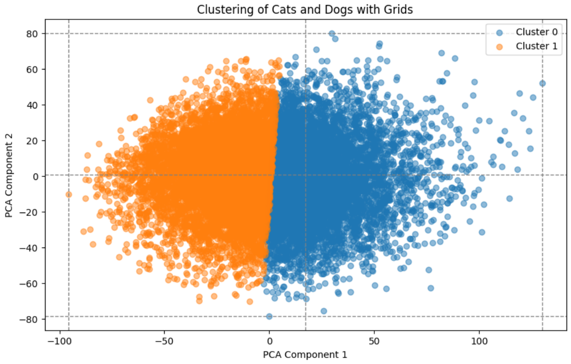

Fig. 8. Cat and Dog Cluster Grids

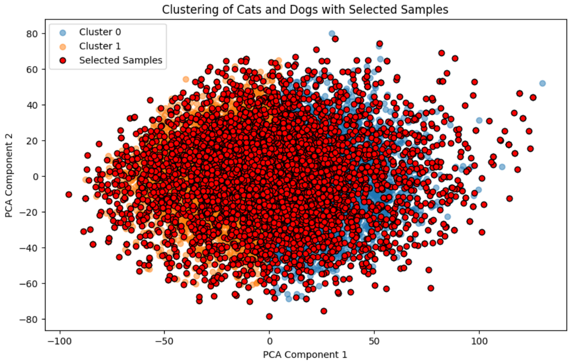

Fig. 9. Cat and Dog Selected Points via Grid Cells

As can be seen, this method gives nearly identical results to our base model with only including 10% of the dataset. Slicing the cluster into grids and choosing examples from each closed area makes the data distribution more normalized and achieves a similar feature representation of the original dataset. By increasing the number of grids, better results may be achievable in larger datasets.

D. Data Augmentation Distillation

Data augmentation is a technique used to artificially increase the size of a dataset by applying various transformations to the existing data samples. This helps improve the generalization capability of machine learning models and reduces overfitting. Here is the theoretical approach for the data augmentation method:

1) **Determination of Augmentation Ratio**: Specify the augmentation ratio, which determines the proportion of new samples to be generated relative to the original dataset size.
2) **Random Selection of Samples**: Randomly select a subset of samples from the original dataset for augmentation. The number of samples to be augmented is calculated based on the augmentation ratio.
3) **Application of Augmentation Techniques**: For each selected sample, apply a random augmentation technique chosen from a predefined set of transformations. Common augmentation techniques include:
    - Horizontal or vertical flipping
    - Rotation within a certain range
    - Gaussian blurring
    - Adjusting brightness or contrast
4) **Generation of Augmented Data**: Apply the selected augmentation technique to each sample to create new augmented samples. Preserve the original labels for the augmented samples.
5) **Verification and Model Training**: Verify the shapes of the augmented data and labels to ensure consistency with the original dataset. Finally, use the augmented dataset to train machine learning models, such as convolutional neural networks, for improved performance on downstream tasks.

By applying data augmentation, we can effectively increase the size and diversity of the training dataset, leading to more robust and generalizable machine learning models.

```
# Augmentation Distillation
Test accuracy: 0.7706666588783264
Test loss: 0.8599461317062378
```

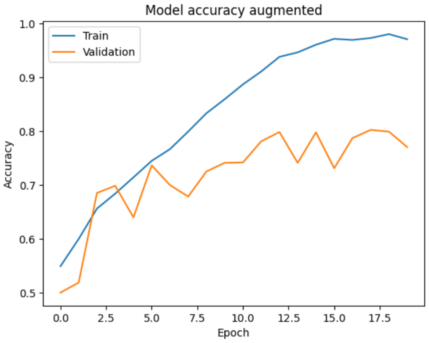

Fig. 10. Cat and Dog Cluster

By creating a synthetic dataset from a small portion of the original dataset, we tried to train our model on this set. It achieved an acceptable result however, it is not the best compared to other methods.

V. Data

The dataset contains 7,500 cat images and 7,500 dog images, making a total of 15,000 pictures. Each subset has an equal number of cat and dog images, showing a fair mix of both types. These images come from a larger dataset with millions of pictures, but we picked 15,000 to manage computational costs. While the sizes of these images may vary, we resize them all to 150x150 pixels for consistency in our model. Additionally, we include some corrupted images to help the model handle tough situations. We also keep aside 10% of the images for testing our model's performance. This diverse and manageable dataset helps us develop and check robust machine learning models for accurate cat and dog image sorting.[3]

Fig. 11. Cat 1

Fig. 12. Cat 2

Fig. 13. Dog 1

Fig. 14. Dog 2

VI. Results

The performance of the three distillation methods—random sampling, clustering, and data augmentation—was compared against the base model to evaluate their effectiveness in reducing dataset size while maintaining classification accuracy. The base model, trained on the full subset of the original dataset, achieved an accuracy of 91%.

Random distillation showed varied results across different sampling percentages. With 50% of the data, the model achieved an accuracy of 74%, while 25% and 10% data resulted in accuracies of 81% and 75%, respectively. Despite some variations, random sampling demonstrated the potential to maintain reasonable accuracy with reduced data sizes.

Clustering distillation proved to be highly effective, achieving an accuracy of 89%, which is comparable to the base model. This method successfully preserved the essential features of the dataset by selecting diverse samples from cluster grids, ensuring a balanced representation of the original data.

Data augmentation distillation, on the other hand, achieved an accuracy of 77%. While this approach increased the diversity of the training data, it did not outperform the clustering method but still maintained a satisfactory level of accuracy.

Overall, clustering distillation emerged as the most effective method, providing a balance between dataset size reduction and maintaining high classification accuracy.

VII. Conclusion

This study explored the effectiveness of different dataset distillation methods—random sampling, clustering, and data augmentation—in reducing the LSUN cat and dog dataset size while preserving model performance. By reducing the dataset subset to just 10% of its original size, we aimed to alleviate computational costs without significantly compromising accuracy.

The clustering distillation method proved to be the most promising, closely matching the base model's performance. Random sampling showed potential but with variable results and data augmentation provided a satisfactory outcome, demonstrating the robustness of synthetic data generation.

The clustering method of dividing the space and selecting evenly distributed examples from each grid is proven to be very effective compared to randomly selecting examples. Future applications and varying techniques of this process may be implemented.

These findings suggest that efficient dataset distillation can make large datasets more manageable, allowing for quicker training times and reduced computational resources while maintaining high accuracy. Future work could explore combining these methods or applying them to different datasets to further enhance their utility in practical applications.

References

[1] S. Panigrahi, A. Nanda and T. Swarnkar, ”Deep Learning Approach for Image Classification,” 2018 2nd International Conference on Data Science and Business Analytics (ICDSBA), Changsha, China, 2018, pp. 511-516, doi: 10.1109/ICDSBA.2018.00101.

[2] Samuel Cortinhas, https://www.kaggle.com/code/samuelcortinhas/mnist-dataset-distillation

[3] Dataset, https://www.kaggle.com/datasets/serdarbicici1/petimagescat-and-dog