Final Homework Write-up

Manisha Subrahmanya
Computer Engineering Major
Texas A&M University
College Station, TX 77845
manisha.is@tamu.edu

Abstract

This project develops a multimodal classification model that utilizes both visual and audio data to predict digit labels from 0 to 9. The dataset consists of pairs of grayscale images and corresponding audio snippets. To address this challenge, a neural network with separate convolutional pathways for each modality was designed, integrating extracted features through concatenation followed by a dense classification network. The model was trained and validated, focusing on optimizing the F1 score (macro-average) to account for class imbalances. Initial results demonstrate the model's effectiveness in leveraging multimodal data to enhance predictive accuracy, suggesting robust generalization capabilities for diverse test scenarios.

1 Introduction

The challenge in this project is to effectively combine visual and audio data to enhance the accuracy of digit classification. Digit recognition is a foundational task in the field of machine learning, commonly applied in scenarios ranging from automated data entry systems to assistive technologies. Traditionally, digit recognition has heavily relied on visual data. However, by incorporating audio data that corresponds to spoken digits, the robustness and reliability of the classification process can be significantly improved, especially in noisy or visually obstructed environments.

2 The Method

2.1 Data Preprocessing

The data is available on the course Kaggle page here The dataset used in this project consists of multimodal data, specifically images and audio files, each corresponding to digits from 0 to 9.

Effective preprocessing is crucial for optimizing the performance of machine learning models. The chosen methods aim to standardize the input data, enhance the feature quality, and make the training process more efficient and robust. Here's a detailed breakdown of the preprocessing techniques applied to the multimodal dataset and the reasoning behind each choice.

VISUAL DATA
Each image was resized to 28x28 pixels to standardize the input size for the neural network, ensuring uniform contribution across all images and optimal compatibility with the model architecture. Pixel values were normalized to a range between 0 and 1 by dividing by 255, the maximum possible value for a grayscale image. This normalization reduces data variability, speeds up the learning process, prevents the model from focusing excessively on outliers, and reduces the risk of gradient explosions during training.

AUDIO DATA
Signals were transformed into spectrograms using the Short-Time Fourier Transform (STFT). This

method converts audio into a time-frequency representation, making it easier for convolutional networks to detect patterns and recognize spoken digits by capturing both temporal dynamics and frequency content. The spectrogram values were then normalized by dividing each by the maximum value found across the dataset. This ensures that all audio data is on a comparable scale, similar to the image normalization, enhancing the stability and efficiency of the network's learning process.

2.2 Model Design

A custom Multimodal Neural Network was developed to process and integrate visual and audio data effectively. This network comprises separate pathways for each modality, which are then combined before the classification stage. This design ensures modality-specific processing, allowing for optimized feature extraction from both images and audio independently before their integration. This method enhances prediction accuracy by utilizing the complementary information from both modalities, which is particularly beneficial in scenarios where data from one modality might be ambiguous or less informative.

The architecture of the model includes distinct pathways:

Visual Pathway: This consists of convolutional layers, ideal for processing image data, equipped with batch normalization and max pooling. These layers capture spatial hierarchies and normalize layer outputs for stability, reducing feature map dimensions efficiently.

Audio Pathway: Tailored for audio data, this pathway uses 1D convolutional layers to capture temporal audio patterns. Like the visual pathway, it includes batch normalization and max pooling for effective feature processing.

Classifier: Post feature integration from both modalities, the network transitions into fully connected layers. These layers, combined with dropout, mitigate overfitting risks and finalize predictions.

2.3 Model Training

To train the multimodal neural network effectively for digit classification using visual and audio data, a structured approach was adopted using the PyTorch framework, leveraging its dynamic computation graphs and efficient GPU acceleration.

Data Preparation and Splitting: The dataset underwent standard preprocessing, including resizing images to 28x28 pixels, converting audio signals into spectrograms, and normalizing all inputs to facilitate uniform learning. Data was divided into an 80-20 split between training and validation sets to ensure the model could be evaluated on unseen data.

Training Process: Training was conducted in batches of 32 to optimize memory and computational efficiency. The network was trained over 20 epochs, where each epoch represented a complete pass through the training data.

Loss Function and Optimization: A weighted cross-entropy loss function was used to account for class imbalances, prioritizing minority classes to prevent bias. The Adam optimizer was chosen for its effective handling of sparse gradients and adaptive learning rate adjustments, essential for deep neural network training.

Validation and Performance Monitoring: After each training epoch, the model was evaluated on the validation set using the macro-averaged F1 score, which assesses precision and recall across all classes. This metric is particularly valuable for imbalanced datasets.

Adjustments and Tuning: Performance metrics such as loss and F1 score were monitored to guide modifications in learning rates and network architecture, ensuring optimal learning and preventing issues like overfitting.

This meticulous training approach enabled the model to learn effectively from both data modalities, enhancing its prediction accuracy and ensuring robust generalization capabilities for diverse test scenarios.

2.4 Hyperparameter Tuning

The parameters for the model are set based on standard practices and preliminary testing, which include:

- **Number of Layers:** The model uses two convolutional layers in each pathway before flattening and combining. This depth was chosen to balance model complexity and computational efficiency.
- **Number of Neurons/Filters:** Each convolutional layer in the visual pathway starts with 32 filters and is doubled in the subsequent layer to 64. This progression helps in capturing more complex features progressively. The audio pathway mirrors this structure.
- **Kernel Size:** A kernel size of 3x3 for the visual layers and 3 for the 1D audio layers was selected to capture local patterns effectively.
- **Pooling:** Max pooling with a window of 2x2 (2 for 1D) is used to reduce the dimensionality after each convolutional layer.
- **Dropout Rate:** A dropout rate of 0.5 was used after the fully connected layers to mitigate overfitting.
- **Learning Rate:** Set to 0.0005, a conservative value to start training without causing large updates that might skip optimal solutions.

To fine-tune these parameters and explore others, the following strategies were employed:

1. **Grid Search**
   For some key hyperparameters, such as learning rate and batch size, a grid search was conducted. This involves:
   - Training the model with various combinations of these parameters.
   - Evaluating each combination on the validation set using the F1 score.
   - Selecting the combination that yields the best performance.

   This method is exhaustive but effective for exploring a manageable number of hyperparameter combinations.
2. **Random Search**
   Given the high computational cost of grid search for more extensive hyperparameter sets, random search was employed to explore other parameters like the number of layers and dropout rates. This method randomly selects combinations within defined ranges and is more efficient for discovering good configurations in a large search space.
3. **Manual Tuning**
   Based on insights from both grid and random searches, further refinements were made manually:
   - Adjustments to the architecture, such as increasing or decreasing the number of neurons in the layers based on the model's performance and the complexity of features being learned from both modalities.
   - Incremental adjustments to learning rate based on training dynamics observed (e.g., if the model was learning too slowly or too quickly).

3 Results

The results can be found in 'submissions.csv'.

4 Conclusion

REASONS FOR PERFORMANCE

- **Dual-Pathway Architecture:** The choice to process visual and audio data separately before combining them allowed for specialized handling of each data type, optimizing feature extraction. This approach likely contributed to the model's ability to utilize the strengths of both modalities, leading to improved accuracy.

- **Preprocessing:** Standardizing images to a uniform size and scale and converting audio into a consistent format (e.g., spectrograms) ensured that the input data was in an optimal form for the network to process. This consistency likely helped in reducing noise and enhancing the model's learning efficiency.
- **Integration of Modalities:** Combining features from both pathways before the classification stage helped the model make more informed predictions by leveraging complementary information, which is particularly beneficial in scenarios where one modality may not be as clear or informative.

AREAS FOR IMPROVEMENT

Despite these strengths, there's always room for improvement. Here are potential steps to enhance the model's performance:

- **Deeper or Alternative Network Architectures:** Experimenting with deeper networks or different architectures such as ResNets for images and recurrent neural networks (RNNs) or transformers for audio could capture more complex patterns in the data.
- **Advanced Feature Engineering:** For audio, exploring more sophisticated feature extraction methods like Mel-frequency cepstral coefficients (MFCCs) or chroma features might yield richer information. For images, techniques like data augmentation (rotations, shifts, zooms) could make the model more robust to variations in the visual data.
- **Hyperparameter Optimization:** More extensive hyperparameter tuning, possibly using automated methods like Bayesian optimization, could fine-tune the learning rate, batch size, number of layers, and dropout rates to better suit the specific characteristics of the data.
- **Regularization Techniques:** Incorporating additional regularization techniques such as L1/L2 regularization, or exploring alternative forms of dropout, could help in preventing overfitting, especially if the network architecture becomes deeper.
- **Ensemble Methods:** Deploying ensemble techniques, where multiple models are trained independently and their predictions are combined, could lead to more robust predictions. This approach often improves performance as it aggregates diverse perspectives on the data.
- **Cross-Validation:** Implementing k-fold cross-validation during training would provide a more rigorous validation of the model's performance across different subsets of the data, enhancing its generalizability.
- **Post-Processing of Predictions:** Implementing advanced decision-making logic in the final prediction stage, possibly incorporating rules based on the confidence of predictions or the characteristics of misclassified examples, could refine the outputs.