U.PORTO

FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

Faculdade de Ciências da Universidade do Porto

Final Report

Deep Learning Classifiers for Urban Sound Data

Margarida Vila Chã, 202107923
Sebastião Santos Lessa, 202103238
Alexandre Marques, 202106956
5th of November 2023, Porto

Índice

1. Introduction

Description of the classification problem considered

In this project, we tackle the urban sound data classification problem, a task with significant real-world applications. The urban sound data classification problem addressed here involves categorizing sound excerpts into one of ten predefined classes:

- Air Conditioner
- Car Horn
- Children Playing
- Dog Bark
- Drilling
- Engine Idling
- Gun Shot
- Jackhammer
- Siren
- Street Music

The objective is to develop deep learning classifiers capable of accurately determining the class to which a given sound excerpt belongs. Categorizing sounds in urban environments is a complex task due to the diverse and nuanced nature of sound sources. Accurate classification is crucial for various applications, including noise monitoring, urban planning, and public safety. This project focuses on leveraging advanced deep learning techniques to enhance the precision and efficiency of urban sound classification, contributing to the broader field of audio pattern recognition.

The Practical Impact of Urban Sound Classification: Applications and Implications

Urban sound classification studies offer a myriad of practical applications, significantly impacting various aspects of our daily lives. By leveraging advanced deep learning techniques, the insights gained from these studies contribute to several critical domains:

1. Noise Monitoring and Urban Planning:

Precise classification of urban sounds is fundamental for effective noise monitoring in metropolitan areas. City planners can utilize this information to implement targeted strategies for noise reduction, thereby enhancing the overall quality of life for urban residents. The acoustic landscape analysis aids in urban planning, ensuring that noise-sensitive zones, such as residential areas or educational institutions, are appropriately considered during the planning and development phases.

2. Public Safety and Emergency Response:

The rapid and accurate recognition of emergency sounds, such as sirens or gunshots, is paramount for public safety. Deep learning classifiers applied to urban sound data can facilitate timely responses from law enforcement and emergency services. This technology enhances crisis management efforts, potentially saving lives in critical situations.

3. Smarter Cities and Improved Quality of Life:

The outcomes of urban sound classification studies contribute to the creation of smarter cities. Understanding the acoustic environment enables the optimization of infrastructure development, fostering environments that are not only more efficient but also more livable. Ultimately, these advancements aim to improve the overall quality of life for urban residents.

In summary, the applications of urban sound classification extend beyond academic research, playing a vital role in shaping the future of urban living, noise management, public safety, and city planning.

2. Description of the deep learning solutions considered for the problem

In tackling the urban sound classification problem, our approach involved the utilization of two powerful deep learning architectures:

- Multi-Layer Perceptron (MLP);
- Convolutional Neural Network (CNN).

Multi-Layer Perceptron (MLP)

The Multi-Layer Perceptron (MLP) is a fundamental architecture in the domain of artificial neural networks. As a feedforward neural network, the MLP comprises multiple layers of nodes, each layer fully connected to the next. It excels in learning complex, non-linear relationships within data. The network architecture involves an input layer, one or more hidden layers, and an output layer. Neurons in each layer utilize activation functions, such as the rectified linear unit (ReLU), to introduce non-linearity, thereby enhancing the model's capacity to capture intricate patterns in the data. The MLP is versatile and well-suited for various classification tasks.

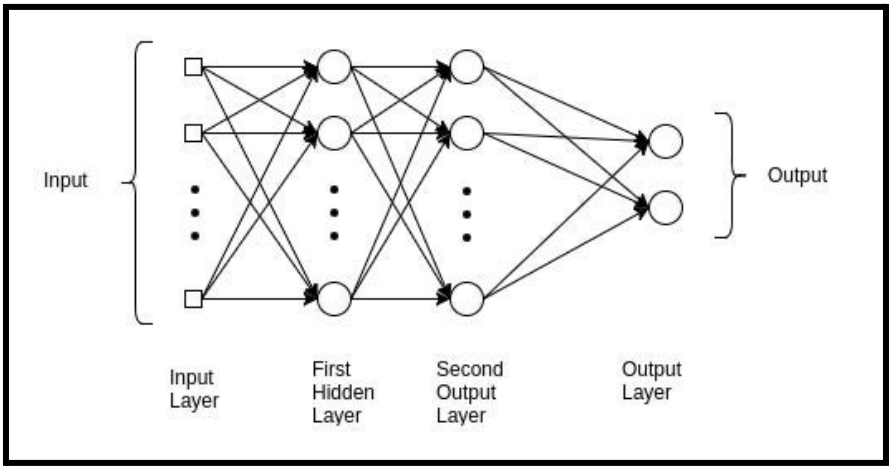

Image 1: Example of Multi-Layer Perceptron

Convolutional Neural Network (CNN)

The Convolutional Neural Network (CNN) is a specialized type of neural network designed for processing structured grid data, such as images or, in this case, spectrograms. CNNs are particularly adept at capturing spatial hierarchies within data, making them powerful tools for image and pattern recognition tasks. The distinctive feature of CNNs is the use of convolutional layers, which systematically apply convolutional operations to input data. This architecture allows the network to automatically learn and extract hierarchical features from the input. CNNs have demonstrated remarkable success in image and audio classification tasks, making them a popular choice for various deep learning applications.

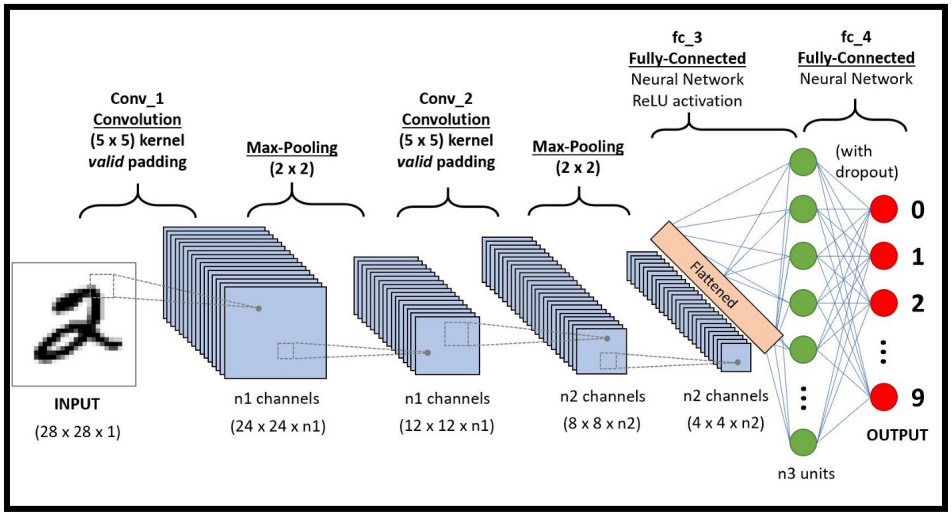

Image 2: Overview of Convolutional Neural Network

3. Implementation

Language and IDE used

The implementation of the urban sound classification project was carried out using Python as the programming language. The primary integrated development environment (IDE) utilized for this project was Jupyter Notebook, providing an interactive and collaborative environment for code development and experimentation.

Two distinct classifiers were employed for the urban sound classification task: a Multi-Layer Perceptron (MLP) and a Convolutional Neural Network (CNN).

Pre-Processing and Data Preparation

The pre-processing pipeline involved extracting relevant features from raw sound data using the Librosa library. To address variations in durations and sampling rates, the data was normalized. Object values in the dataset were converted to numeric form, with consideration given to handling irregularities in the data. Features were then standardized using the StandardScaler.

Model Design

Multi-Layer Perceptron (MLP)

The MLP classifier was designed with consideration of the number of layers, neurons per layer, and activation functions. The architecture consisted of two hidden layers, each employing the rectified linear unit (ReLU) activation function. The number of neurons per layer was determined through experimentation, incorporating 512 neurons. The final layer utilized the softmax activation function to produce class probabilities and has an output of 10 neurons.

- **Implementation**: The MLP classifier was implemented using the TensorFlow and Keras libraries. The model was trained on the pre-processed and resampled training data. The performance was evaluated on the test set, considering accuracy as the primary metric.

Convolutional Neural Network (CNN)

The CNN architecture was tailored to capture spatial patterns within the input features. 1D inputs and convolutional layers with rectified linear unit (ReLU) activations were employed, followed by max-pooling 1d layers for spatial downsampling. The final layers included a

flattening operation to prepare the data for dense layers, ultimately leading to the output layer with softmax activation for multi-class classification.

- **Implementation**: The CNN was implemented using the TensorFlow and Keras libraries. The architecture incorporated convolutional layers with max-pooling, followed by dense layers. The model was trained on the reshaped and scaled training data. Evaluation on the test set involved computing accuracy, confusion matrix, and a comprehensive classification report.

Training strategies adopted

Choice of Parameters:

The choice of parameters is fundamental for optimizing the performance and generalization of our machine learning models. In our project, we selected parameters such as the number of layers in the neural network, layer sizes, and other values that significantly influence the architecture and behavior of the model. This process ensures that our models are tailored to capture the intricate patterns within the urban sound data.

- ### Learning Rate:

  The learning rate is a critical factor influencing the convergence and stability of training. In our implementation, we chose a learning rate of 0.0005, striking a balance to ensure steady convergence while preventing overshooting during training.

- ### Mini-Batch Size:

  The batch size, affecting computational efficiency and training stability, was chosen judiciously. A batch size of 128 was employed for MLP and 32 for CNN, optimizing the trade-off between computational efficiency and the smoothness of the optimization process.

- ### Number of Epochs:

  The number of epochs, representing how many times the model goes through the entire dataset during training, was set at 30. This decision was guided by monitoring of the model's convergence over time, ensuring that an adequate number of epochs were employed for effective learning without risking overfitting.

Dataset Utilization and Cross-validation Strategy:

The selection and preparation of the dataset constitute a critical aspect of our model development, as we leveraged the Urbansound8k dataset comprising 8732 labeled sound

excerpts across ten classes. In alignment with best practices, we adopted the 10-fold cross-validation technique to ensure the reliability and generalization capacity of our models.

The 10-fold cross-validation process involves systematically using each folder for training and testing. During each iteration, nine folds are utilized for training, while one fold is exclusively reserved for testing- this is done 10 times, where each time a different folder is used. This meticulous approach guarantees that every predefined folder serves as the test set exactly once and is employed nine times for training throughout the entire cross-validation process.

Our commitment to this comprehensive methodology stems from the understanding that it allows us to capture a more accurate representation of our models' performance across diverse data distributions. Importantly, it mitigates potential biases associated with a static training-test split. By consistently leveraging the pre-defined folds in the urbansound8k dataset, we m

aintain methodological consistency in our evaluation, showcasing the models' ability to generalize effectively across distinct subsets of the data.

This robust cross-validation methodology provides a trustworthy estimation of our models' performance, instilling confidence in the reported results and reinforcing the reliability of our urban sound classification models.

Choice of Optimizer:

The optimizer is a key component in searching for the best model weights. We opted for the Adam optimizer, considering its advantages in terms of convergence, training speed, and adaptability to different architectures. This choice aligns with our goal of achieving efficient and effective optimization since we got better results with it compared to SGD.

Regularization Techniques:

Mitigating overfitting is crucial for model generalization. We employed regularization techniques, including early stopping, dropout, data augmentation and L1L2 weight regularization, to enhance the models' ability to generalize well to unseen data. These techniques contribute to the overall robustness of our models.

Transfer Learning:

In the context of urban sound classification, where the dataset is relatively specific and might have distinct features compared to general sound datasets, employing transfer learning might not provide substantial benefits. Transfer learning is often more effective when the source and target domains share similar characteristics. The decision to train models from scratch can be justified by the uniqueness of the urban sound classification problem and the dataset used.

We opted to not use transfer learning due to the fact that the model needs to be task specific to the data we are trying to predict and we had a large amount of data to train a model from scratch. Besides that, some models need to have a specific input shape that was not compatible with our features’ shape.

Early Stopping:

The early stopping technique is crucial for preventing overfitting and speeding up training. In our case, we chose to monitor the validation loss during training. The early stopping criterion was set to halt training if there was no improvement in validation loss after 20 consecutive epochs for the MLP model. In the case of the CNN architecture, the patience parameter was set at 5. This strategy aimed to ensure that the model generalizes well to new, unseen data while avoiding unnecessary computational expenses.

Weight Regularization:

For the chosen architectures (MLP and CNN), there were cases where the complexity of the models, coupled with dropout layers, seemed sufficient to prevent overfitting. After experimenting, we observed that a low number for the regularizer was the best choice and we got the best results with 0.001 for the MLP model. For the case of the CNN we obtained better results when not incorporating weight regularization.

Dropout:

Dropout is a specific regularization technique employed to improve the generalization of the model. In both the MLP and CNN architectures, dropout layers were strategically incorporated. These layers randomly deactivate a certain percentage of neurons during training, preventing reliance on specific neurons and enhancing the network's ability to generalize to new data. Dropout proved effective in mitigating overfitting and improving the robustness of our models, improving our accuracy and getting the best score when at 0.5.

Data Augmentation:

Data augmentation is useful for helping improve model generalization. Since we saw a significant difference in some Label’s count, which could lead the model to struggle learning patterns from the minority class, we decided to address class imbalances with oversampling, using the Synthetic Minority Over-sampling Technique (SMOTE) on the features of the testing folders.

We will explore the chosen data augmentation techniques and how they contribute to the model's robustness.

4. Results

The evaluation of our urban sound classification models, employing both the Multi-Layer Perceptron (MLP) and Convolutional Neural Network (CNN), reveals interesting insights into their performance across different sound categories.

Accuracy Comparison:

1. **Multi-Layer Perceptron (MLP):**
    - The MLP model exhibits varying degrees of accuracy across different sound categories, ranging from 48.42% to 65.51%.
    - Notably, the MLP demonstrates particularly strong performance in folders 4 and 8, achieving accuracies of 61.11% and 65.51%, respectively.
    - However, there are folders, such as 2 and 3, where the MLP's accuracy is comparatively lower, standing at 48.42% and 48.43%.

2. **Convolutional Neural Network (CNN):**
    - The CNN model demonstrates a different accuracy distribution across the sound categories, with values ranging from 53.50% to 62.29%.
    - Folders 5 and 10 stand out as strengths for the CNN, achieving accuracies of 62.29% and 59.62%, respectively.
    - While the CNN performs consistently well across most folders, there is room for improvement in folders 2, 3, and 8, where accuracies are relatively lower.

Comparative Analysis:

1. **MLP vs. CNN:**
    - The comparative analysis highlights the nuanced strengths of each model. The MLP excels in certain sound categories, showcasing its adaptability, while the CNN demonstrates a more consistent performance across a broader spectrum.
    - It is essential to consider the specific requirements of the application when selecting between the MLP and CNN, as their strengths and weaknesses may align differently with the desired outcomes.

2. **Graphs:**
   Interpretation of Generated Graphs:
   The generated graphs provide a comprehensive view of the performance of the Multi-Layer Perceptron (MLP) across different folds in the 10-fold cross-validation process. Here's how you can interpret the graphs:

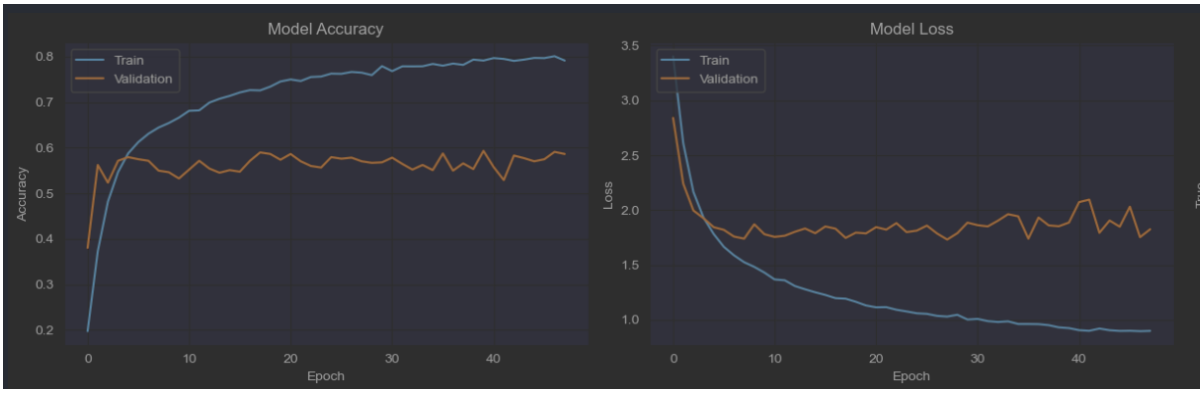

Image 3: Example of graphs from the notebook

1. **Accuracy vs. Epochs:**
    - The first graph, titled "Model Accuracy," depicts the evolution of accuracy on both the training and validation datasets over epochs.
    - If the training accuracy continues to increase while the validation accuracy decreases, it suggests overfitting. Conversely, if both accuracies increase but converge, it indicates a well-generalized model.
2. **Loss vs. Epochs:**
    - The second graph, titled "Model Loss," illustrates the training and validation loss trends over epochs.
    - A decreasing training loss and increasing validation loss might indicate overfitting. Conversely, a steady decrease in both losses suggests effective learning and generalization.
3. **Confusion Matrix:**
    - The third graph, a confusion matrix, visualizes the model's predictions against true labels for each class.
    - A strong diagonal line with high values indicates accurate predictions, while off-diagonal elements represent misclassifications.
    - The heatmap color intensity reflects the quantity of predictions, aiding in identifying which classes the model struggles with.

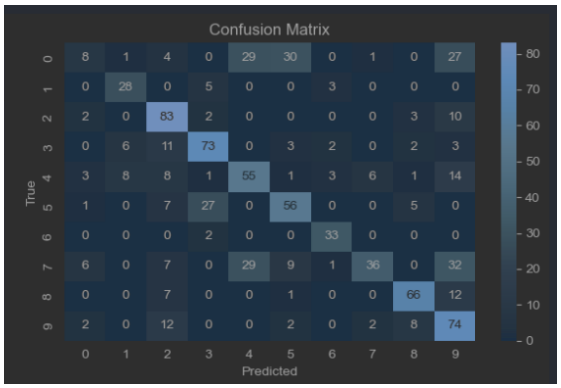

Image 4: Example of Confusion Matrix from the Notebook

In summary, the graphs serve as valuable tools for understanding model behavior, identifying potential issues like overfitting, and assessing overall classification performance.

4.3 Future Considerations:

1. **Fine-Tuning and Optimization:**
    - Further investigation into hyperparameter tuning and model optimization could enhance the overall accuracy of both the MLP and CNN.
2. **Limiting the number of Features and Data Augmentation:**
    - The choice of only using some features and exploration of some other data augmentation techniques might further enhance the models' ability to generalize to diverse sound patterns.

5. Conclusion and Discussion

In reflecting on our project, we find satisfaction in the practical insights gained through the implementation of both a Multi-Layer Perceptron (MLP) and a Convolutional Neural Network (CNN). While our achieved accuracy might not stand out, the making of the project itself has been immensely valuable.

In particular, the CNN posed unique challenges, primarily revolving around input shapes. This aspect prompted deeper research and exploration compared to the relatively smoother MLP implementation. Despite the lower numerical accuracy, we contend that our decision-making and implementation approach were well-founded. We would have liked to have a better accuracy, however we were not able to achieve it.

The most significant obstacle in our project was the meticulous process of parameter tuning for each neural network. Initially, we attempted a grid search; however, the considerable computational power required, led us to abandon this method. Instead, we resorted to a combination of online research and hands-on experimentation, manually tweaking parameters to achieve a better performance.

In conclusion, this hands-on experience has sharpened our problem-solving and critical thinking skills. The intricacies of neural network development are now clearer, and we come out of this project with a greater understanding of the decision-making required to create a successful machine learning model.

6. Bibliographic References

- OpenAI. Chat with GPT-3.5. https://chat.openai.com. Acesso durante todo o desenvolvimento do trabalho.
- Moodle. (n.d.). [Aprendizagem Computacional II]. https://moodle2324.up.pt/course/view.php?id=2262 . Acesso durante todo o desenvolvimento do trabalho.
- TechTarget. (n.d.). Neural Network Definition. https://www.techtarget.com/searchenterpriseai/definition/neural-network Acesso na parte inicial do desenvolvimento do projeto.
- GitHub. (n.d.). Basic Multi-Layer Perceptron Implementation. https://github.com/jorgesleonel/Multilayer-Perceptron/blob/master/Basic%20Multi-Layer%20Perceptron.ipynb Acesso na parte inicial do desenvolvimento do projeto.
- Nogueira A. , Oliveira H. , Machado J. , Tavares J. (8 November 2022). Sound Classification and Processing of Urban Environments: A Systematic Literature Review. Sensors,https://web.fe.up.pt/~tavares/downloads/publications/artigos/sensors-22-08608.pdf#page=28&zoom=100,48,584 Acesso na parte inicial do desenvolvimento do projeto.
- Mishra M. (2020, August 26). Convolutional Neural Networks Explained. Towards Data Science. https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939 Acesso na parte inicial do desenvolvimento do projeto.

- Biswal A. (Nov 7, 2023) .Convolutional Neural Network Tutorial. Simplilearn. https://www.simplilearn.com/tutorials/deep-learning-tutorial/convolutional-neural-network Acesso na parte inicial do desenvolvimento do projeto.