Deployment Report

Questions and Answers

Q: Does the model perform as accurately as expected on your smartphone? List a few methods to improve the model's accuracy.

A: The model's performance on the smartphone does not entirely match the expected accuracy. Some methods to improve the model's accuracy include:

- Increasing the amount of training data.
- Fine-tuning the model with transfer learning.
- Using data augmentation techniques to create more diverse training samples.
- Optimizing the hyperparameters for better performance.

Q: When building a model for resource-limited hardware, how do you balance fast inference times with acceptable model accuracy? What trade-offs did you encounter?

A: Balancing fast inference times with acceptable model accuracy involves optimizing the model to reduce complexity while maintaining performance. Trade-offs encountered include:

- Reducing the model size, which may lead to a slight decrease in accuracy.
- Pruning unnecessary layers and neurons to speed up inference.
- Quantizing the model to lower precision, which accelerates processing but may affect accuracy slightly.

Screenshots

[Screenshot: Screenshot (161).png]

[Screenshot: Screenshot (162).png]

[Screenshot: Screenshot (163).png]

[Screenshot: Screenshot (164).png]

[Screenshot: Screenshot (165).png]

Videos

Demonstration videos:

- [Keyword-spotting model working on smartphone](https://photos.app.goo.gl/1N2RhHNKZrC2CWzR9)

Reflections

Deploying the model to the smartphone and Arduino board was a learning experience. Some technical difficulties included managing memory constraints on the embedded board and optimizing the model for real-time inference. Observations included the model's sensitivity to noise and the importance of consistent data preprocessing for reliable performance. Compile time was longer due to the complexity of the model, but it recognized the keywords effectively once deployed.