Development Report

Lukas Matulaitis (w20036800)

Department of Computer and Informational Sciences

Northumbria University

Newcastle upon Tyne, United Kingdom

[W20036800@nothumbria.ac.uk](mailto:W20036800@nothumbria.ac.uk)

# Baseline Pipeline

The Initial version of the dataset prepared and tested on were phone recordings of multiple keypresses of several keys on the keyboard.

# Baseline Solution

Baseline solution is provided within the "main" branch of the GitHub repository. For iterative development we moved to the branch "CNN-Test".

**Disclaimer:**

I was having trouble detecting touch peaks of keypresses reliably, so for the method "detect\_touch\_peaks", I received help from my teammate Aron Bencsik.

# Baseline Evaluation

For detecting touch peaks, I used an approach suggested by Cecconello S et. al. in the paper "Skype & Type: Keyboard Eavesdropping in Voice-over-IP"

<https://dl.acm.org/doi/abs/10.1145/3365366?casa_token=Vq1IvygqxWMAAAAA:Z_716satRhemFDm72DIn7zYipcc0MnJIQcWWhLkpCsh5zFQHuCzWuxDq-gs8YtTjBX0OpdHmDsLb2Q>

Using this approach, the audio is divided into 10ms segments, and FFT magnitudes are summed within each of them. This essentially provides the loudness of each 10ms segment, which is then used to detect touch peaks in the recorded audio.

Detecting keypresses for feature extraction works reliably but needs manual fine-tuning for different recordings. This is due to the fact that some keypresses can be louder or quieter than others, so the threshold at which a peak should be considered a touch peak needs to be adjusted manually so that all the touch peaks get recognized, but none of the release peaks get detected as touch peaks. Furthermore, there is a time frame after a touch peak, during which the algorithm does not try to detect touch peaks to avoid detecting release peaks. But this time frame variable may also need to be manually fine-tuned depending on the recordings used.

mfcc feature extraction, labelling of the extracted features, and providing a mapping for the labels (e.g. label 1, mapping a) and saving them to a table for training works extremely reliably.

A simple deep neural network designed reaches accuracies of 35%-50%, which is not satisfactory.

# Conclusion

All in all, the baseline solution implemented is modular enough to use with ease in the group's

iterative development without needing the be changed internally, with the possible exception of

detecting touch peaks, which could be made more robust to radical changes in key tap loudness.

Furthermore, the architecture of the network used needs to be changed.