DECODING KEYBOARD STROKES WITH SOUND AND LANGUAGE

Kudupudi Mohan Kumar

M.Tech (25548)
Computational and Data Sciences
Indian Institute of Science Bangalore, India
Email: mohankk@iisc.ac.in

Dayita Chaudhuri

M.Tech (25338)
Computational and Data Sciences
Indian Institute of Science Bangalore, India
Email: dayitac@iisc.ac.in

ABSTRACT

In this work we implement Deep Learning-based Acoustic Side Channel Attack (ASCA) to classify keystrokes. We first implement a CoAtNet[1] classifier, achieving 93.9% accuracy in individual keystroke recognition. Further our evaluation show that naively concatenating these predictions for sentences yields only 15% accuracy at the sentence level. To address this gap, we introduce a two-stage pipeline that integrates instruction-tuned large language models via few-shot prompting or lightweight LoRA[2] fine-tuning. The approach involves two main steps: first, classifying individual keystrokes using CoAtNet. Secondly, employing the language model to improve the performance of the predicted keystrokes in a meaningful English sentence. This approach boosts sentence-level inference by more than 5x, for both few-shot prompting on 8B models and fine-tuned 3B model, demonstrating that compact, targeted fine-tuning can rival larger models in contextual error correction. This work not only showcases a practical security concern but also contributes to the relevance of language models in enhancing the performance of deep learning systems in real-world applications.

Index Terms— Acoustic side-channel attacks (ASCAs), CoAtNet, Large Language Models

1. PROBLEM DEFINITION AND MOTIVATION

The rise of ubiquitous recording devices, such as smartphones and laptops with built-in microphones, has introduced novel security risks, including acoustic side channel attacks. These attacks exploit incidental sounds—like those produced by typing—to infer sensitive information, such as passwords, without direct access to a target device. Firstly, the baseline paper's[3] approach of using CoATNet to predict keystrokes using the acoustics data achieves close to 95% accuracy on keystrokes but lacks any experimentation with natural language sentences. Through this project, we aim to test the baseline model on sentence data and check the feasibility of providing contextual correlations where classification error creeps in, by integrating a language model and leveraging the statistical properties of natural language.

2. PRIOR WORK

Acoustic side-channel attacks (ASCAs) exploit incidental keyboard sounds to infer sensitive inputs, posing growing security risks in modern settings. Early work relied on statistical and signal-processing techniques[4, 5, 6]. Traditional machine-learning methods like HMMs[7], K-means clustering[8] and SVMs[9] have been used to enhance keystroke classification but struggle with diverse typing styles, noisy environments, and full-text recovery tasks[10].

Recent studies have used Deep Neural Networks leveraging convolutaions and transformers blocks to achieve as high accuracies. CoAtNet [1] introduced a hybrid convolution-attention model, excelling in image classification and applicable to keystroke audio spectrograms. Harrison et al. [3] developed a transformer-based ASCA pipeline using CoAtNet, achieving 95% accuracy on phone-recorded keystrokes, though it lacked contextual error correction. Park et al. [11] enhanced this by integrating vision transformers for classification and large language models (LLMs) for error correction, improving accuracy in noisy settings. Their use of LLMs to refine predictions informs our project's approach to boosting ASCA accuracy through advanced language modeling, building on these foundational contributions.

3. PROGRESS

In this work, we adopt CoAtNet [1] as our keystroke classifier from the baseline paper[3]. CoAtNet fuses convolutional layers with transformer-style self-attention to capture both local and global patterns in mel-spectrogram inputs. Trained on Zoom-recorded keystroke sounds spanning 67 key classes, the model achieves 93.9%accuracy in individual key classification, closely matching the 95% baseline. To extend beyond isolated keystrokes, we integrate instruction-tuned LLMs—Llama 3.2 (1B, 3B, 8B) and Mistral 7B—via few-shot prompting and LoRA fine-tuning. This two-stage

pipeline first predicts raw key sequences, then refines them contextually, boosting sentence-level accuracy by up to four-fold compared to naïve assembly, and demonstrating that lightweight fine-tuning on synthetic noisy–clean text pairs can rival larger models in error correction.

4. EXPERIMENTS AND RESULTS

We use the baseline paper dataset with 41 classes (26 alphabet, 10 numeric characters and 5 additional keys), with 25 keystrokes per key, recorded on Zoom platform. We use Nvidia RTX A6000 to train our CoAtNet model and fine-tune the language models.

4.1. Data Processing and Augmentation

We use STFT to compute energy and use adaptive thresholding to isolate keystrokes, followed by grouping by proximity to avoid over-splitting. For testing on sentence where number of keystrokes is not know, we threshold at the $95th$ percentile. Keystrokes are converted to mel spectrograms using 64 mel bands, a window length of 1024 samples and hop length of 500 resulting in 64x64 images. Prior to feature extraction, signals are time-shifted randomly by up to 40% in either direction. The spectograms were masked using random 10% of both the time and frequency axis and setting all values within those ranges to the mean.

4.2. CoAtNet vs ViT Performance

We used the C-C-T-T CoAtNet Model described in **??**. We trained the CoAtNet model on this augmented dataset, achieving a classification accuracy of 93.86% for individual keystrokes. We also experimented with pre-trained ViT (Vision Transformers) and achieved accuracy of less than 50% due to less data availability. Hence we concluded that CoAtNet is a better alternative for our use case.

4.3. Audio Spectrogram Transformer (AST):

The Audio Spectrogram Transformer (AST) is a Vision Transformer architecture pre-trained on mel-spectogram images. It treats mel-spectrograms as 2D "images" and splitting them into patch embeddings. It processes these patches through multi-head self-attention layers to capture both temporal and spectral dependencies across the entire input. When we use this AST model to finetune it on the keystroke classification task. It achived an accurcacy of 92% on the dataset.

4.4. Performance Improvement with Language Models

Our experiments revealed that the model's performance is still low, close to 15% while predicting the text in a meaningful sentence, depicting that some error creeps in for every sentence, making it very hard to predict the entire sentence correctly. To address this, we integrated a language model into the system. We experimented with Llama 3.2 (1B, 3B, 8B and Mistral 7B models. First, we used few-shot prompting to prompt the instruction-tuned LLMS to correct errors in the predicted sentence. We achieved better sentence accuracies with 7B and 8B models, moderate values with 3B and lower accuracies with 1B model. Additionally, we fine-tuned the 1B and 3B Llama3 models using LoRA[2] on a synthetically generated noisy-clean sentence data from the WikiText corpus[12], allowing it to understand the statistical properties of natural language. We use the noisy sentence predictions of CoAtNet and prompt the language model to get the final corrected predictions. This combination yields higher accuracy in predicting the text typed on a keyboard, demonstrating the effectiveness of language models in this context, refer to Fig.2. We find that lighter models with targeted fine-tuning show comparable performance with their heavier counterparts.

5. SUMMARY OF CONTRIBUTIONS AND NOVELTY

In this paper, implement the baseline CoAtNet model for acoustic side-channel attacks (ASCAs) and evaluate it at the sentence level. While prior work demonstrated high accuracy in classifying individual keystrokes (94%), we show that naïvely concatenating these predictions into natural language yields only 15% sentence accuracy. To address this gap, we introduce a two-stage pipeline that couples a CoAtNet-based keystroke classifier with large language models for contextual error correction. This integration boosts sentence-level inference by up to four times compared to keystroke-only assembly.

Moreover, we demonstrate that lightweight, targeted fine-tuning with LoRA on synthetic noisy-clean sentence pairs generated from the WikiText-2 corpus enables 1 B and 3 B parameter Llama models to match the zero-shot performance of their larger counterparts. Through comprehensive benchmarking on different metrics (refer figure 1) we quantify the trade-offs between model size, compute cost, and final sentence accuracy.

6. RESPONSIBILITY OF INDIVIDUAL MEMBERS

- **Kudupudi Mohan Kumar:** Conducted literature review; Performed data collection and data processing pipeline design. Implemented and compared deep-learning, CoAtNet, and Vision Transformer architectures; Trained CoAtNet to high keystroke-classification accuracy. Finetuned the AST model on dataset.
- **Dayita Chaudhuri:** Performed literature survey, data-processing pipeline design, and keystroke segmentation; built a custom dataset pipeline for LLM fine-tuning; integrated LLMs via few-shot prompting and LoRA fine-tuning.
- **Collaboration:** Jointly defined research goals, exchanged experimental insights, and co-authored the

final report and presentation.

7. REFERENCES

[1] Zihang Dai, Hanxiao Liu, Quoc V Le, and Mingxing Tan, "Coatnet: Marrying convolution and attention for all data sizes," *Advances in neural information processing systems*, vol. 34, pp. 3965–3977, 2021.

[2] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al., "Lora: Low-rank adaptation of large language models.," *ICLR*, vol. 1, no. 2, pp. 3, 2022.

[3] Joshua Harrison, Ehsan Toreini, and Maryam Mehrnezhad, "A practical deep learning-based acoustic side channel attack on keyboards," in *2023 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE, 2023, pp. 270–280.

[4] Ximing Liu, Yingjiu Li, Robert H Deng, Bing Chang, and Shujun Li, "When human cognitive modeling meets pins: User-independent inter-keystroke timing attacks," *Computers & Security*, vol. 80, pp. 90–107, 2019.

[5] Sourav Panda, Yuanzhen Liu, Gerhard Petrus Hancke, and Umair Mujtaba Qureshi, "Behavioral acoustic emanations: Attack and verification of pin entry using keypress sounds," *Sensors*, vol. 20, no. 11, pp. 3015, 2020.

[6] Tong Zhu, Qiang Ma, Shanfeng Zhang, and Yunhao Liu, "Context-free attacks using keyboard acoustic emanations," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2014, pp. 453–464, ACM.

[7] Li Zhuang, Feng Zhou, and J. Doug Tygar, "Keyboard acoustic emanations revisited," *ACM Transactions on Information and System Security (TISSEC)*, vol. 13, no. 1, pp. 1–26, 2009.

[8] Esan Wit and Thijs Houtenbos, "All your keystrokes are belong to us: A foray into the world of acoustic keylogging," Technical report, Offensive Technologies Project, Jun 2014, Available at https://www.academia.edu/35190735/Esan$_matthijs(accessed$: $2025 - 04 - 28)$.

[9] Jian Wang, Rukhsana Ruby, Lu Wang, and Kaishun Wu, "Accurate combined keystrokes detection using acoustic signals," in *2016 12th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN)*. 2016, pp. 9–14, IEEE.

[10] Alireza Taheritajar, Zahra Mahmoudpour Harris, and Reza Rahaeimehr, "A survey on acoustic side channel attacks on keyboards," in *International Conference on Information and Communications Security*. 2024, pp. 99–121, Springer.

[11] Jin Hyun Park, Seyyed Ali Ayati, and Yichen Cai, "Improving acoustic side-channel attacks on keyboards using transformers and large language models," *arXiv preprint arXiv:2502.09782*, 2025.

[12] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher, "Pointer sentinel mixture models," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

8. SUPPLEMENTARY DIAGRAMS AND CHARTS

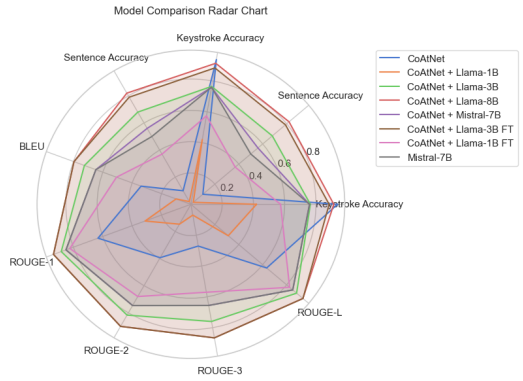

Fig. 1. Comparison of Models over different metrics.

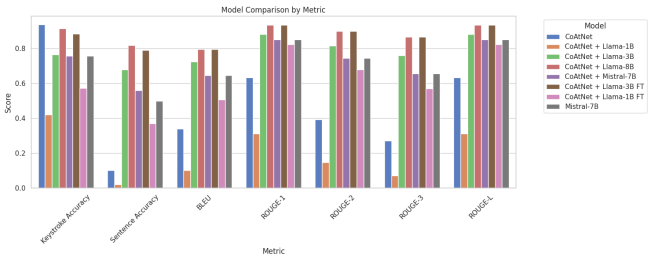

Fig. 2. Performance of different Model combinations.

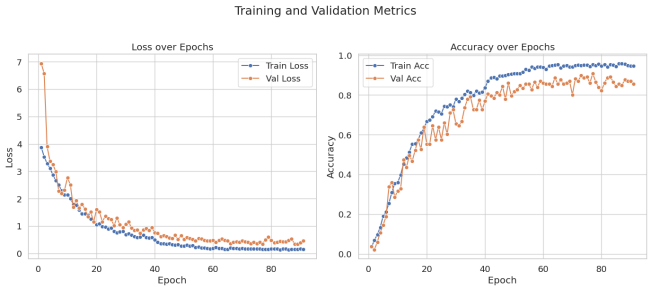

Fig. 3. Training and Validation Metrics for CoAtNet

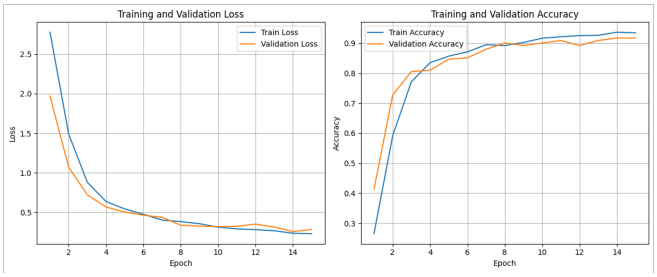

Fig. 4. Training and Validation Metrics for AST