Making Acoustic Side-Channel Attacks on Noisy Keyboards Viable with LLM-Assisted Spectrograms' "Typo" Correction

Seyyed Ali Ayati[1] Jin Hyun Park[1] Yichen Cai[2] Marcus Botacin[1]

[1]Texas A&M University
[2]University of Toronto

USENIX WOOT'25
Seattle, WA

ali.a@tamu.edu

TEXAS A&M UNIVERSITY
Department of Computer Science & Engineering

Agenda

Is This a Good Password?

`Lw7@NcQhZ#f8GvXsT2rY`

Passwords Are Gone!

- This is a strong password.
- But can you remember it? Most people can't.
- So we switched to something better. . .

Figure 1: Generated by AI

What About Passphrases?

`this is a strong password`

Enter The Passphrase!

`this is a strong password`

- Passphrases are easier to remember and just as strong.
- They've become the best practice for humans typing passwords.
- Problem solved? Not quite...

Metric	**Passphrase** `this is a strong password`	**Random Password** `Lw7@NcQhZ#f8GvXsT2rY`
Charset Size	58	94
Shannon Entropy	86.49 bits	86.44 bits
Combinations	1.22×10^{44}	2.90×10^{39}

Table 1: Entropy comparison of a passhphrase and a random password.

Both are equally secure![1]

[1] Ref: https://alecmccutcheon.github.io/Password-Entropy-Calculator/

Passphrases are strong and memorable, but are they truly secure in every environment?

They Can Still Be Stolen

Figure 2: Generated by AI

- Even your passphrase isn't safe if someone is listening.
- Acoustic Side-Channel Attacks (ASCAs) exploit the sound of your keyboard.

The Attacker's Setup Is Simple

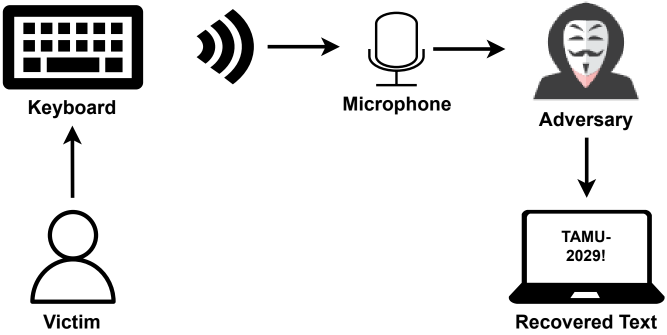


Figure 3: The attacker doesn't need access to your device. Just a recording—from a call or a nearby phone.

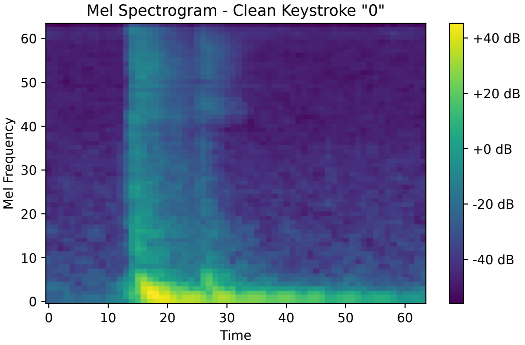
If the setup is simple, what's stopping attackers from succeeding in real-world scenarios?

But...

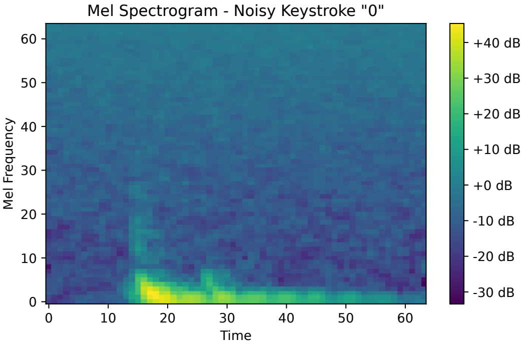
Figure 4: ASCAs on keyboards always had one big weakness: noise. [2]

[2] AI-Generated

Noise Destroys The Features



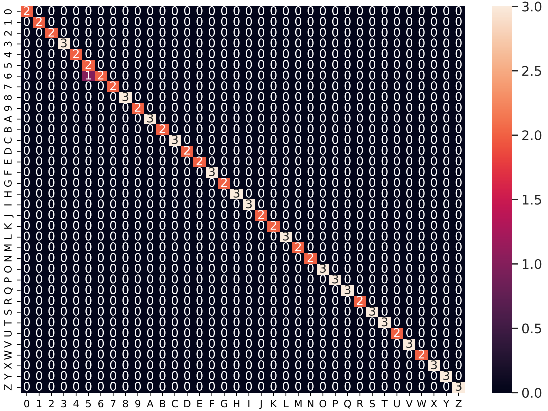
(a) Clean Spectrogram



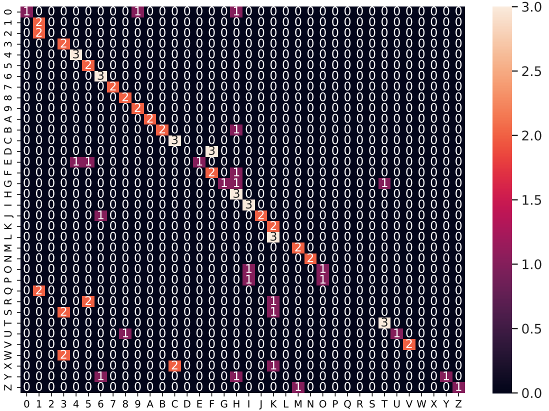
(b) Noisy Spectrogram

Figure 5: (a) and (b) show clean and noisy spectrograms, respectively. Light noise (10%) masks the distinctive keystroke patterns.

Even the Best Models Fail Under Noise



(a) CoAtNet on clean audio (**99%** accuracy)



(b) CoAtNet on noisy audio (**58%** accuracy)

Figure 6: (a) and (b) show the confusion matrices of CoAtNet on clean and noisy audio, respectively.

What if we change the model? Will VTs help?
If the best models struggle, how can we make ASCAs viable in noisy conditions?

Split the Problem

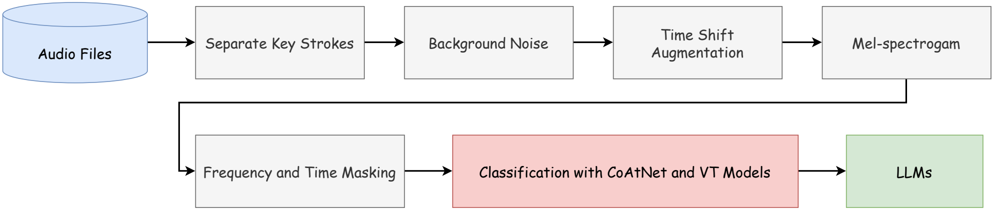


Figure 7: After preprocessing, a Vision Transformer (e.g., Swin) or CoAtNet classifies individual keystroke spectrograms, producing a sequence of noisy character predictions; then, a Large Language Model (e.g., GPT-4o or LLaMA) corrects transcription errors to produce clean, intelligible text.

Before and After: The LLM Fixes It

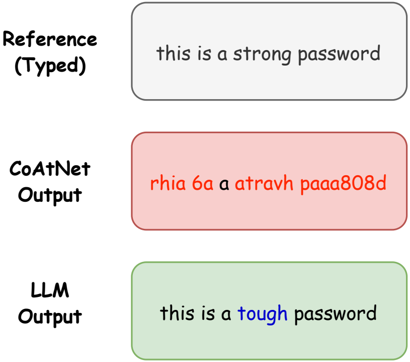


Figure 8: The initial text sequence (top) represents the ideal output. The noisy prediction (bottom) introduces typographical and semantic errors due to environmental noise or model inaccuracies.

Stage 1: Spectrogram $\rightarrow$ Character (Keystroke Detection)

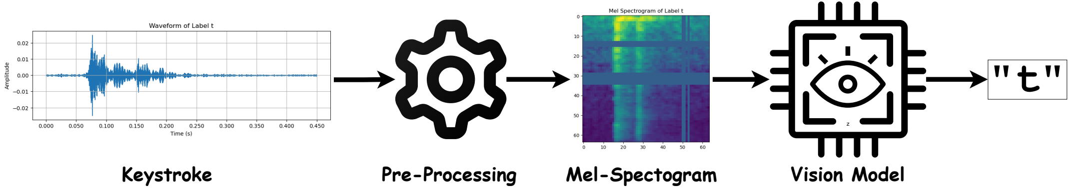


Figure 9: Stage 1

- **Pre-Processing:** Time-shift, time/freq masking data augmentation, transforming into 64x64 mel-spectrogram.
- **Vision Models:** CoAtNet (Baseline), Vision Transformers (ViT, Swin, DeiT, CLIP, BEiT)
- **Datasets:** Zoom, Phone (36 keys $\times$ 25)

Stage 1: Key Results

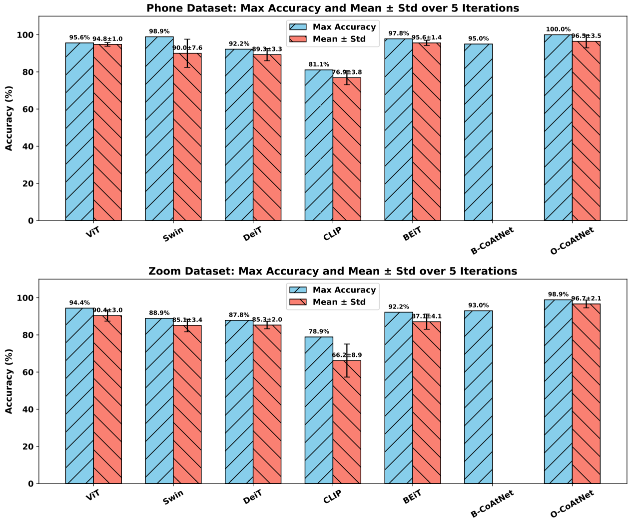


Figure 10: This plot compares max and average accuracies of different models on the Phone dataset. Max values are in blue, mean ± std in red.

Highlights:

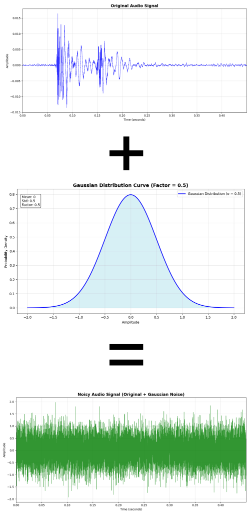
- O-CoAtNet achieves highest scores.
- CLIP consistently underperforms.
- BEiT and Swin show strong accuracy.

Stage 2: LLM for Context-Aware Correction

- Gaussian noise is added to the dataset at varying levels.
- Models make mistakes (accuracy drops) due to the noise.
- The output sequences often contain typos and, in some cases, are unreadable.
- Can an LLM fix this?

Table 2: Noise factor (η) applied to each dataset: low, medium, and high correspond to approximately 10%, 20%, and 50% accuracy reductions, respectively.

Dataset	Noise factor (η)		
	Low	Medium	High
Phone	0.012	0.024	0.06
Zoom	0.1	0.5	1.0



Stage 2: LLM Prompt Structure

System Role:
You are an expert in correcting typos in sentences.

User Role:
Here are pairs of sentences with typos; learn from them:

`sentence:` $\{\ S^1_{\texttt{pred}}\ \}$
`corrected:` $\{\ S^1_{\texttt{true}}\ \}$

`sentence:` $\{\ S^2_{\texttt{pred}}\ \}$
`corrected:` $\{\ S^2_{\texttt{true}}\ \}$

Now, please correct these sentences and output only the corrected version with no additional text:
$\{\ S_{\texttt{pred}}\ \}$

Intuition

This few-shot prompt guides the LLM to learn correction patterns and apply them to a new input.

Stage 2: Evaluation Metrics

BLEU

Measures **precision** of overlapping n-grams (1–4) between model output and reference. Penalizes overly short outputs.

METEOR

Uses **precision + recall** with flexible matching (stems, synonyms, paraphrases) and a penalty for word order differences.

ROUGE-1 / ROUGE-2

Recall-oriented: counts overlapping unigrams (ROUGE-1) or bigrams (ROUGE-2), capturing vocabulary coverage and short phrase accuracy.

ROUGE-L

Based on the **longest common subsequence** between output and reference, reflecting structural similarity and word order.

Stage 2: Key Results

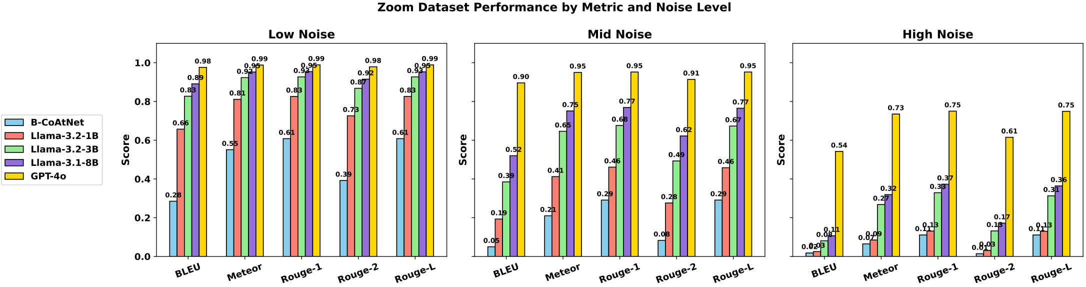


Figure 12: Performance of various models on the Zoom dataset across BLEU, METEOR, and ROUGE metrics under different noise levels. GPT-4o consistently outperforms smaller models, especially under high noise conditions.

GPT-4o outperforms all models. We cannot use it offline, and other big models are resource intensive. Can we achieve the same performance with a much smaller model?

Can Smaller Models Match the Giants?

The Challenge

While **LLaMA-3.1-8B** and **GPT-4o ($\sim 200B$)** achieve state-of-the-art performance, they are:

- Resource-intensive (need 16–30+ GB RAM/VRAM)
- Impractical for low-resource or stealthy attack scenarios
- Expensive to deploy at scale

Our Goal

Investigate whether a much smaller model, like **LLaMA-3.2-3B**, can be:

- Fine-tuned or prompted effectively
- Competitive in performance with large models
- Suitable for practical, real-world attacks

Small Model, Big Results

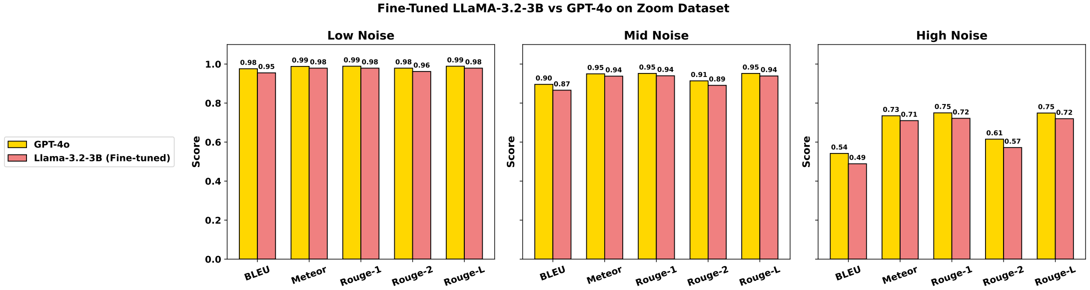


Figure 13: GPT-4o vs fine-tuned LLaMA-3.2-3B (LoRA) across metrics and noise levels on the Zoom dataset.

→ Achieves 90% of GPT-4o performance with 1.5% of its model size.

Limitations & Future Work

Field-Wide Gaps

- **Dataset Availability:** Public ASCA datasets are tiny (36 keys, no space/backspace), limiting what any research can currently test.
- **Noise Realism:** Community datasets lack real-world ambient noise — most use synthetic Gaussian noise as a proxy.
- **Hardware Diversity:** Nearly all public datasets focus on a single device type (e.g., MacBook Pro), making cross-device benchmarks rare.

Call to the Community

- Collaboratively build and release **large, open-access ASCA datasets**.
- Include recordings with realistic ambient noise conditions.
- Cover diverse devices and keyboards to enable true generalization.

Key Takeaways & Conclusion

What We Showed

- **Vision Transformers (VTs)** achieved accuracy close to the current state-of-the-art (CoAtNet), showing strong potential for keystroke spectrogram recognition.
- **LLMs** are essential for handling real-world noise in post-processing.
- **Fine-tuned small models** enable portable and practical attacks.
- First end-to-end **VT + LLM** pipeline for keyboard ASCAs under noise.
- Achieved $>$**95%** text recovery under medium noise with a small model.

Security Implication

Keystrokes can be inferred acoustically—even strong passphrases are vulnerable.
Adopt MFA and biometrics to mitigate this risk.

Questions?

Thank You!

ali.a@tamu.edu | ali-ayati.com

Slides and code: github.com/Botacin-s-Lab/EchoCrypt

ARTIFACT EVALUATED usenix ASSOCIATION AVAILABLE

ARTIFACT EVALUATED usenix ASSOCIATION FUNCTIONAL

ARTIFACT EVALUATED usenix ASSOCIATION REPRODUCED

SCAN ME