Keyboard Acoustic Emanations Revisited

CCS 2005

Li Zhuang, Feng Zhou, J. D. Tygar (UC Berkeley)

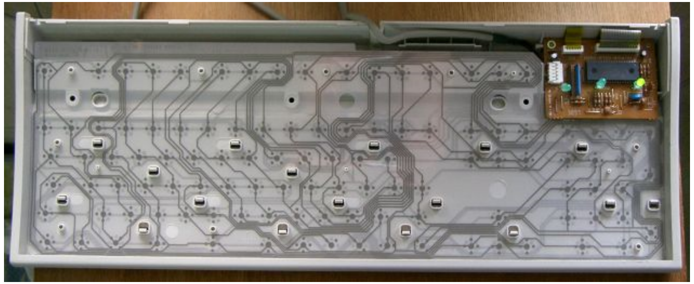
Presented by: Gang Wang

Extracting Information from Side Channels

- Inferring words typed on the keyboard by analyzing the sound

What Is the Intuition?

- Different keystrokes make different sounds
  - Locations
  - Underlying hardware



Takeaway: be sure there exists a pattern before you start “machine learning”

Threat Model and Challenges

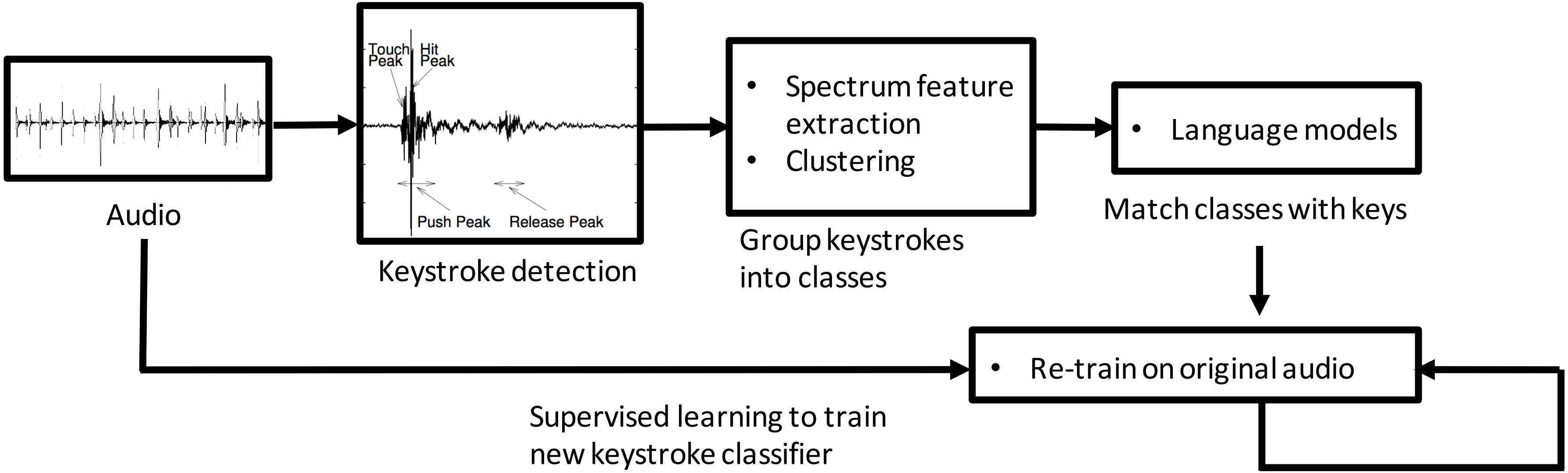
- Attacker has a microphone recording the victim's typing
  - **Assumptions**: typing English text, no labeled input
  - **Goals:** recovering the English text, inferring random text (e.g., password)
- Challenges
  - Hard to obtain labeled training data --- no cooperation from the victim
  - Typing patterns can be keyboard specific
  - Typing patterns can be user specific

Key Intuition: the typed text is often not random.

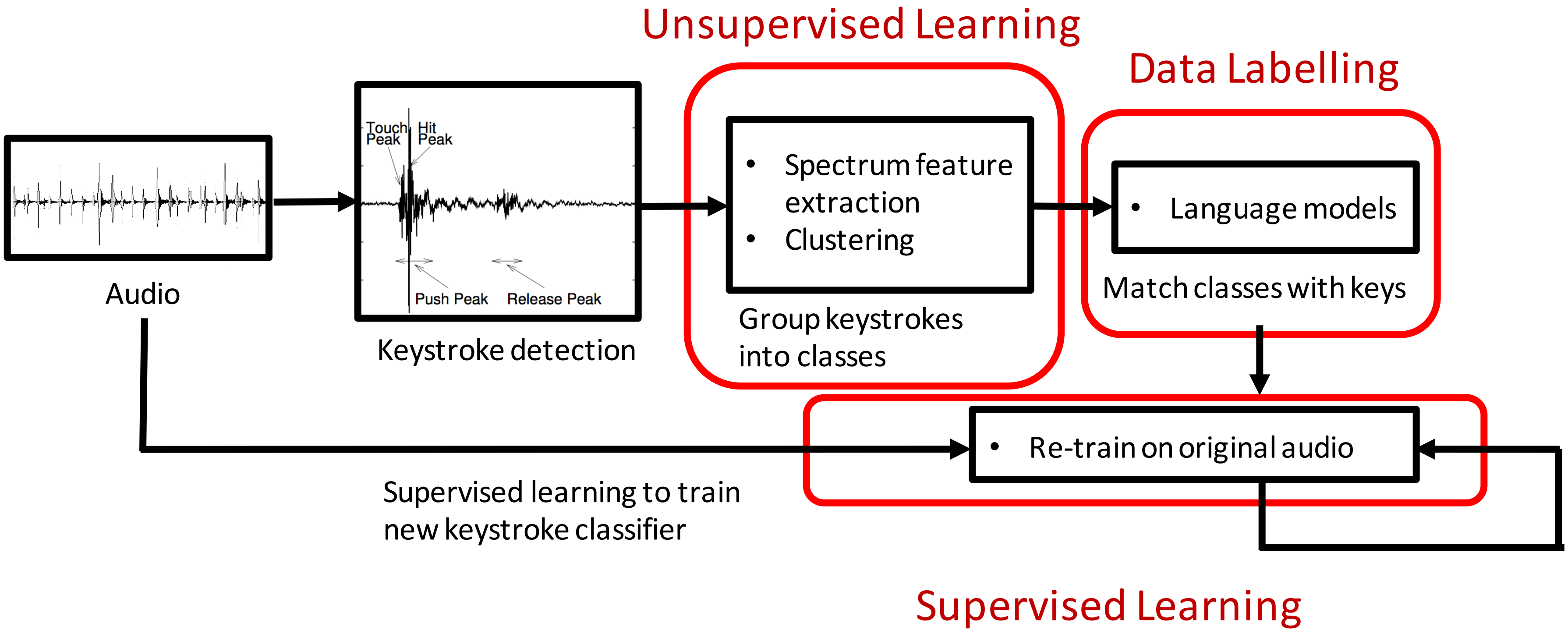
- English words limits the possible temporal combinations of keys
- English grammar limits the word combinations.

How The Attack Works

- Key idea: generating training data automatically
  - Labelling the audio of a key stroke with the actual key



A Combination of Different Learning Methods



Step1: Unsupervised Learning

- Unsupervised clustering
  - Feature generation
    - Cepstrum features
  - Clustering into K classes
    - K > N (actual number of keys used)

- Output
  - K **unlabeled** classes

- Spectrum feature extraction
- Clustering

Group keystrokes into classes

this is the best pizza in town

this is the best pizza in town

Step 2: Context-based Language Model

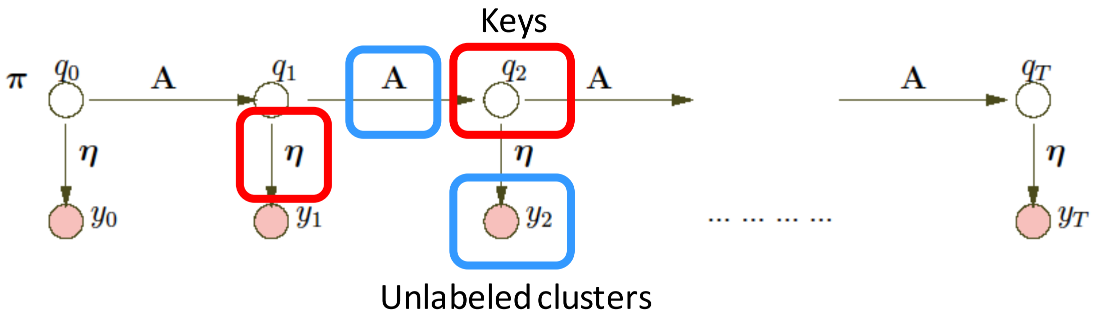
- Need to label the clusters: which key they represent?

- Assume the victim is typing English text
  - Characters follow certain frequency
  - Actual content follows English spelling and grammar

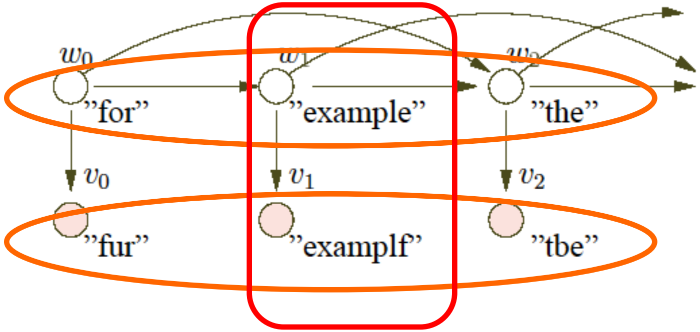
- Advantages:
  - Use 2-character combination frequency to match classes to keys
  - Use language model (spelling, grammar) to correct mistakes

Details: Context-based Language Model

- Character-level mapping:
  - Hidden Markov Model
  - Produce a probability of keys assigned to classes.
  - Example: “th” vs. “tj”



- Word-level correction:
  1. Spell check
  2. Grammar
     - Tri-gram



Details: Context-based Language Model

Before spelling and grammar correction

the big money fight has drawn the shoporo od dosens of companies in the entertainment industry as well as attorneys gnnerals on states, who fear the fild shading softwate will encourage illegal acyivitt, srem the grosth of small arrists and lead to lost cobs and dimished sales tas revenue.

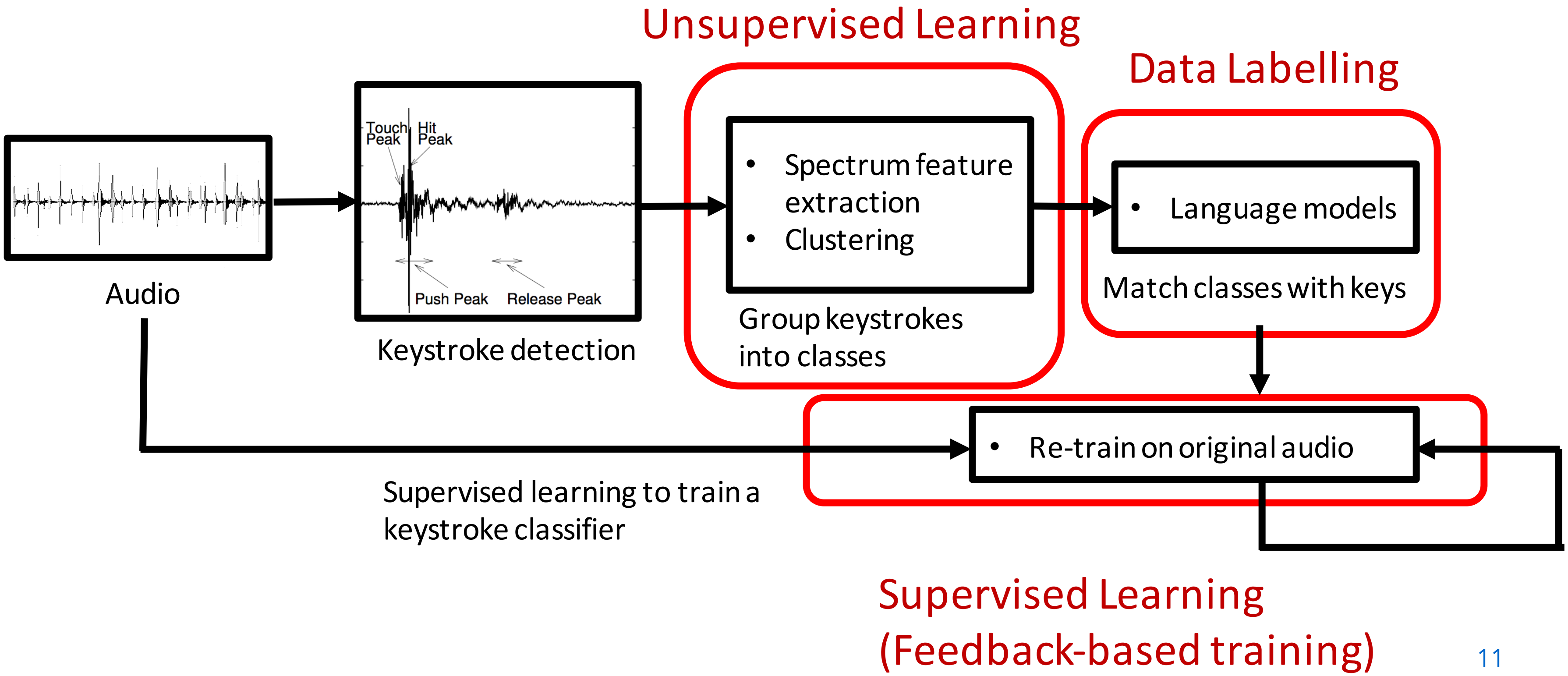
After spelling and grammar correction

the big money fight has drawn the support of dozens of companies in the entertainment industry as well as attorneys generals in states, who fear the film sharing software will encourage illegal activity, stem the growth of small artists and lead to lost jobs and finished sales tax revenue.

_____ = errors in recovery

= errors corrected by grammar

A Combination of Different Learning Methods



Feedback based Training

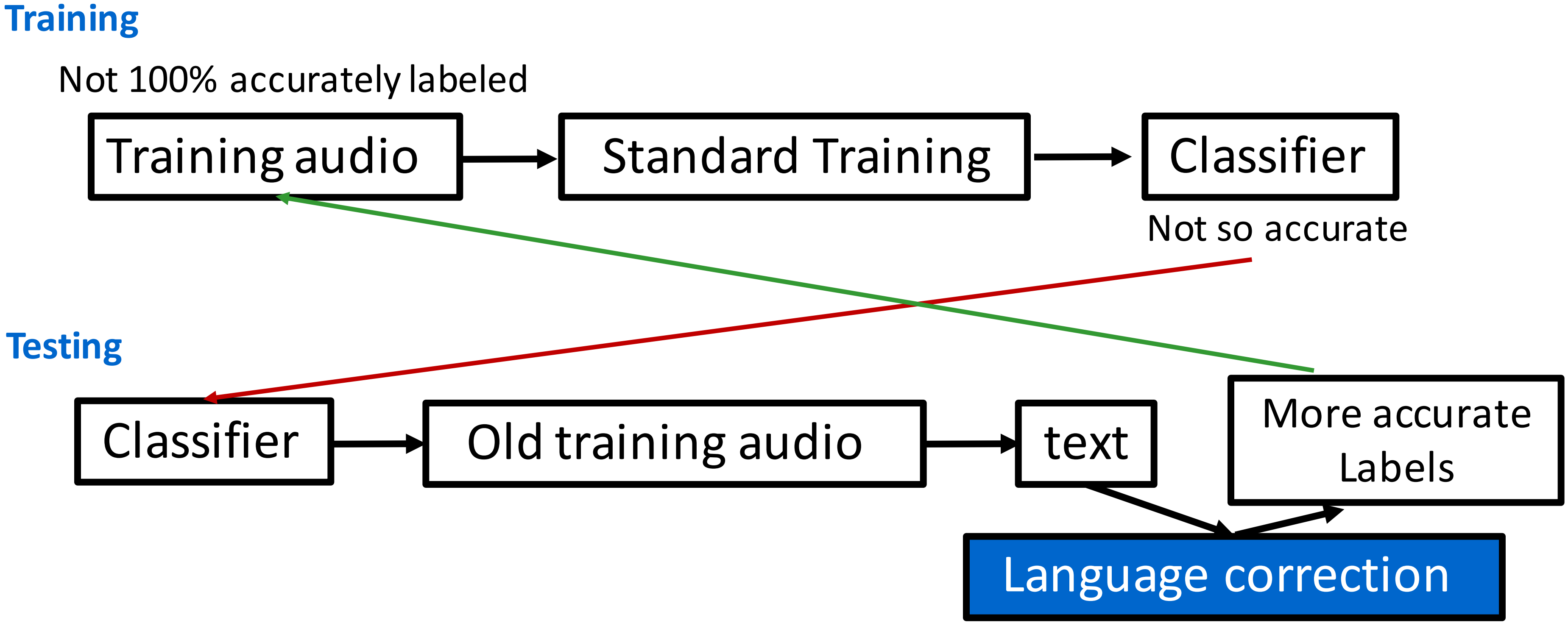
- Re-train on original audio

- A keystroke classifier (for inferring random text)
  - Given a keystroke, produce the label of the key

- Training
  - Input: noisy training data
    - Only a subset of labeled data from the language models
    - Choose those with fewer corrections by the language model (quality indicator)
  - Output: a not so accurate keystroke classifier

- Testing
  - Use the trained classifier to classify the training data again
  - Use the language model to correct the classification result
  - Use the corrected label for re-training

Feedback based Training (Con’t)

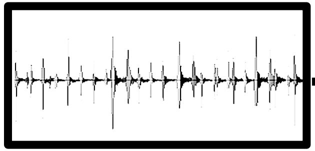


Evaluation

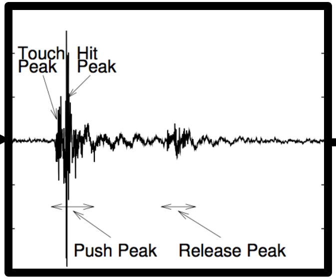
		Set 1		Set 2		Set 3		Set 4	
		words	chars	words	chars	words	chars	words	chars
unsupervised learning	keystrokes	34.72	76.17	38.50	79.60	31.61	72.99	23.22	67.67
	language	74.57	87.19	71.30	87.05	56.57	80.37	51.23	75.07
1st supervised feedback	keystrokes	58.19	89.02	58.20	89.86	51.53	87.37	37.84	82.02
	language	89.73	95.94	88.10	95.64	78.75	92.55	73.22	88.60
2nd supervised feedback	keystrokes	65.28	91.81	62.80	91.07	61.75	90.76	45.36	85.98
	language	90.95	96.46	88.70	95.93	82.74	94.48	78.42	91.49
3rd supervised feedback	keystrokes	66.01	**92.04**	62.70	**91.20**	63.35	**91.21**	48.22	**86.58**
	language	**90.46**	**96.34**	**89.30**	**96.09**	**83.13**	**94.72**	**79.51**	**92.49**

Table 2: Text recovery rate at each step. All numbers are percentages.

Evaluation



Audio



Keystroke Detection

- Spectrum feature extraction
- Clustering

Group keystrokes into classes

Language model only
Keystroke: 37%
Language: 74%

- Language models

Match classes with keys

Sub-set of the output

- Re-train on original audio

Supervised learning to train a keystroke classifier

1st	2nd	3rd
58%	65%	66%
89%	91%	90%

Other Key Results

- Works for random text
  - Inferring passwords that contain English letters only
  - 90% of 5-character random passwords: < 20 attempts
  - 80% of 10-character random passwords: <75 attempts

- Works for multiple types of keyboards

- Even “low-quality” microphones can do the job

Possible Defenses

- Introduce noise into the system
  - Add (random) background noise to key strokes
    - Remove the unique pattern for each key
  - Use quieter keyboards

- Other defenses
  - Two factor authentication (not just typing a password)
  - No microphone in your room?
    - well, your smartphone, your Amazon Alexa

Other Thoughts

- Things that can be improved or "Limitations"
  - 10+ min English content typing
  - No support for numbers or special characters (Backspace, Capslock, Shift)
  - Typing behavior pattern needs to be relatively stable

- Other side-channels
  - Visible light (camera)
  - Hand movements (smart watch)
  - Vibrations (smartphone on your desk)