TẤN CÔNG HỆ THỐNG NHẬN DIỆN GIỌNG NÓI BẰNG CÁCH TẠO MẪU ĐỐI KHÁNG CÓ MỤC TIÊU

Nguyễn Trường Thịnh - 21520110

Tóm tắt

- Lớp: CS519.O11
- Link Github của nhóm: https://github.com/thinhsama/CS519.O11
- Link YouTube video: https://www.youtube.com/channel/UCc9CITQpJzjYnnMZlBQE3NA

Nguyễn Trường Thịnh-21520110

Giới thiệu

- Hiện nay việc chuyển giọng nói thành văn bản là một hướng nghiên cứu thực tiễn giúp tối ưu hóa thời gian gõ văn bản, tích hợp vào các hệ thống tự động như xe tự hành, thiết bị thông minh,...
- Độ chính xác của các mô hình speech-to-text càng cao thì độ tín nhiệm càng nhiều, nên không tránh khỏi việc bị tấn công.
- Với đề tài này, chúng tôi sẽ giới thiệu cách thức tấn công một hệ thống tự động nhận dạng giọng nói (ASR), làm cho đầu ra của hệ thống ASR trả kết quả theo ý muốn attacker mà vẫn qua mặt được người sử dụng chúng.

Mục tiêu

- Tìm hiểu tổng quan bài toán, xây dựng một hệ thống với code hoàn chỉnh.
- Tiếp cận nghiên cứu về xử lý âm thanh, có thể áp dụng một số lý thuyết từ xử lý ảnh vào âm thanh nhưng thách thức hơn.
- Việc tấn công không nhằm đem lại mục đích xấu mà nó làm nền tảng để giúp ích cho các hệ thống ASR tăng cường phòng thủ.
- Đánh giá và phân tích kết quả trên bộ dữ liệu LibriSpeech với gần 1000 giờ audiobooks, bằng cách tấn công hai mô hình ASR nổi tiếng là DeepSpeech và Lingvo phiên bản năm 2019.

Nội dung và Phương pháp

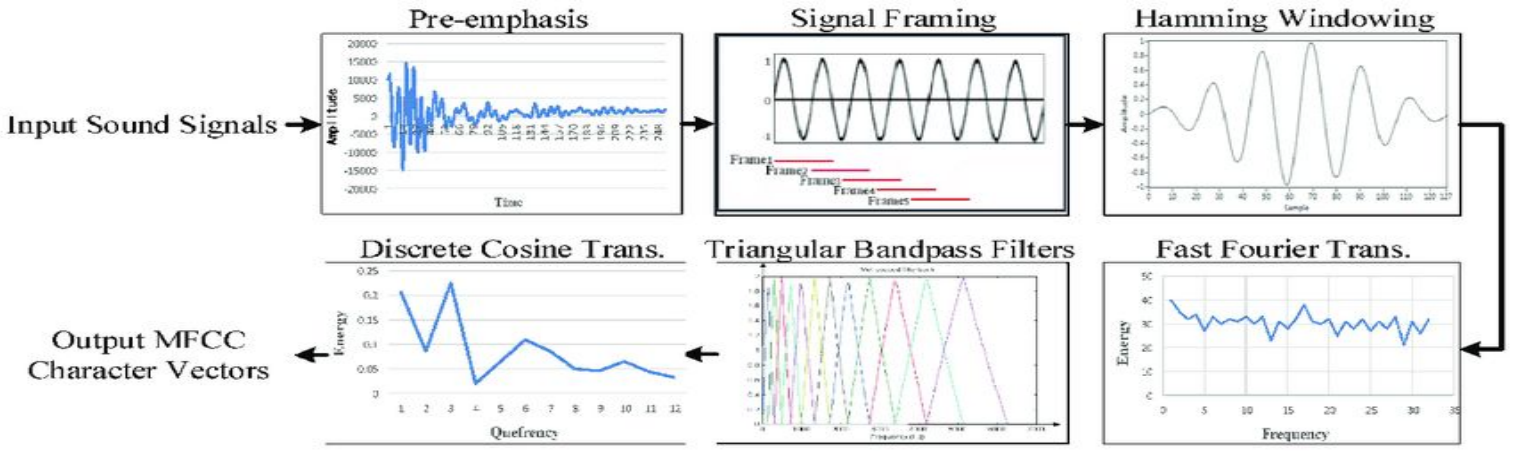
- Nội dung 1: Tìm hiểu tổng quan bài toán:

Tìm hiểu về hệ thống ASR: Cơ bản bao gồm 3 phần acoustic model, lexicon, language model, và một thuật toán tìm kiếm đóng vai trò như decoder. Các phần này có thể kết hợp qua mô hình end-to-end là CTC(Connectionist Temporal Classification) được chúng tôi đề xuất.

Tìm hiểu về cách tạo mẫu tấn công:Trong ngữ cảnh white-box attack, attacker có thể nắm bắt được thông số, thuật toán sử dụng của mô hình ASR, dữ liệu training. Khác với việc đào tạo thông thường là đi tìm trọng số để input x thành output y, ở đây ta tạo ra một input x’ đánh lừa mô hình trả kết quả ra t khác y sao cho x’ có thể đánh lừa người dùng nó là x.

Nội dung và Phương pháp

- Nội dung 2: Tìm hiểu và cài đặt:



Với bước tiền xử lý âm thanh sử dụng: mel-frequency cepstral coefficients(MFCCs).

- Chia tín hiệu âm thanh thành các khung ngắn(khoảng 20-30 mili giây) overlap với nhau.
- Mỗi khung được chuyển đổi sang miền tần số bằng Fast Fourier Transform(FFT)
- Áp dụng bộ lọc Mel-Frequency Filter và dùng Discrete Cosine Transform lên kết quả của filter banks để có được MFCCs.

Nội dung và Phương pháp

- Nội dung 2: Tìm hiểu và cài đặt:

Tìm hiểu và cài đặt phương pháp của Carlini & Wagner và hàm mất mát CTC-loss:

Vì tính chất ngân vang của âm thanh các mô hình nhận diện giọng nói thường chưa trả ra kết quả hoàn chỉnh liền mà trả ra các kết quả “ccaaatt”, “doogg”. Vấn đề này gọi là alignment problem và nó được giải quyết bằng Connectionist Temporal Classification (hay còn viết tắt là CTC). Để cài đặt CTC có thể áp dụng thuật toán quy hoạch động.

Để đi đến công thức của Carlini & Wagner là một phép biến đổi phức tạp nhưng kết quả cuối cùng giống như việc tối ưu hóa dùng gradient descent sau:

$$minimize: \|\delta\|_p + c.f(x+\delta)$$

$$such\ that: x+\delta \in [0,1]^n \quad (2)$$

Trong đó:

- x là mẫu gốc, δ là độ nhiễu loạn
- δ biểu diễn dưới dạng Lp norm.
- f(x+δ)<=0 thì mô hình phân loại sai, theo thực nghiệm ta chọn c nhỏ nhất và f(x+δ)<=0.
- (2) là ràng buộc đảm bảo mẫu đối kháng hợp lệ

Nội dung và Phương pháp

- Nội dung 2: Tìm hiểu và cài đặt:

Tìm hiểu phương pháp ngụy trang bằng mô hình tâm lý âm thanh(psychoacoustic model).

Để đánh lừa tai người psychoacoustic model sử dụng một giả thiết là nhận thức một âm thanh nghe được của tai người có thể bị cản trở khi có sự xuất hiện của một âm thanh khác to hơn (cơ chế auditory masking).

Để đánh giá mô hình chúng tôi dùng độ đo tỷ lệ lỗi từ WER như sau:

$$WER = \frac{S + D + I}{N} = \frac{S + D + I}{S + D + C}$$

Trong đó:

- S là số từ bị thay thế (Substitutions)
- D là số tự bị xóa (Deletions)
- I là số từ bị chèn vào (Insertions)
- C là số từ gốc
- N là số từ ban đầu ($N = S + D + C$)

S+D+I chính là dạng bài toán tính Levenstein distance, là độ đo để đo sự sai khác giữa hai chuỗi khác nhau, hay gặp trong quy hoạch động.

Kết quả dự kiến

- Tài liệu, code và mô tả chi tiết về cách thực hiện bài toán attack adversarial example sử dụng các phương pháp trên.
- Thực hiện tấn công trên hai hệ thống DeepSpeech và Lingvo bằng bộ dữ liệu LibriSpeech với mục tiêu đạt trên 85% tỉ lệ mẫu đối kháng được chuyển thành bản dịch theo mong muốn của Attacker. Đo lường sai khác bằng độ đo WER trên cả trước và sau khi tấn công.
- Lập bảng đánh giá so sánh các mô hình khác nhau với tham số và kết quả đạt được.
- Thực hiện khảo sát mô hình với mọi người để xem khả năng đánh lừa của mô hình trong thực tế và điều kiện môi trường tự nhiên (luôn có nhiễu)

Tài liệu tham khảo

[1]. Nicholas Carlini, David A. Wagner:

Audio Adversarial Examples: Targeted Attacks on Speech-to-Text. IEEE Symposium on Security and Privacy Workshops 2018: 1-7

[2] .Nicholas Carlini, Pratyush Mishra, Tavish Vaidya, Yuankai Zhang, Micah Sherr, Clay Shields, David A. Wagner, Wenchao Zhou:

Hidden Voice Commands. USENIX Security Symposium 2016: 513-530

[3]. Nicholas Carlini, David A. Wagner:

Towards Evaluating the Robustness of Neural Networks. IEEE Symposium on Security and Privacy 2017: 39-57

[4]. Lea Schönherr, Katharina Kohls, Steffen Zeiler, Thorsten Holz, Dorothea Kolossa:

Adversarial Attacks Against Automatic Speech Recognition Systems via Psychoacoustic Hiding. NDSS 2019

[5]. Guangke Chen, Sen Chen, Lingling Fan, Xiaoning Du, Zhe Zhao, Fu Song, Yang Liu:**Who is Real Bob? Adversarial Attacks on Speaker Recognition Systems.** SP 2021: 694-711

[6]. Yao Qin, Nicholas Carlini, Garrison W. Cottrell, Ian J. Goodfellow, Colin Raffel:

Imperceptible, Robust, and Targeted Adversarial Examples for Automatic Speech Recognition. ICML 2019: 5231-5240