

Атака на VAD с помощью сгенерированной музыки

Основная идея работы заключалась в реализации атаки с помощью сгенерированной музыки на систему детекции ключевого слова, которую использует голосовой ассистент Alexa от Amazon.

Для этого была реализована wake-word модель на основе highway блоков, имеющая ту же архитектуру, что и wake-word модель, которая используется Alexa. Модель тренировалась на задачу бинарной классификации: было произнесено ключевое слово “Alexa” или не было.

Атака реализована с помощью алгоритма Карплус-Стронг (генерация мелодии в исполнении струнного инструмента). Алгоритм генерирует мелодию с помощью заданных параметров (бит, частота, мощность). Данный алгоритм был реализован как pytorch nn.Module, что позволило подобрать параметры с помощью градиентной оптимизации. Ключевой момент заключается в том, чтобы перевернуть знак ошибки при оптимизации для увеличения ошибки. То есть параметры для генерации музыки подбираются так, что музыка будет понижать точность системы детекции ключевого слова.

Данные для обучения и теста

Набор данных для обучения и теста состоит из позитивных примеров-аудиодорожек(записано произношение "Alexa") и негативных примеров (любой другой звук/речь/шум).

- Сбор и аугментация позитивных примеров: запись произношения "Alexa" одним человеком в четырех различных вариациях(две на train, две на test), дублирование этого произношения 160(train), 80(test); изменение скорости ($\times 0.85$, $\times 1.15$) и добавление шумов (0.01, 0.02, 0.03) - 200 train, 100 test
- Негативные примеры: датасет LibriSpeech (dev-clean) $\sim (80/20)$

Архитектура модели

- 4 слоя highway блоков для выделения признаков из спектрограмм
- Линейный слой для понижения размерности (bottleneck)
- Временно-контекстное окно для конкатенации соседних фреймов
- 6 слоев highway блоков и линейный слой для задачи классификации

Основные шаги эксперимента:

1. Реализация wake-word модели $f(x)$ на основе highway блоков
2. Реализация Карплус-Стронг алгоритма как `nn.module` с набором параметров `\theta` - назовем его `AudioGenKS`. `AudioGenKS(\theta)`
3. Теперь подберем параметры `\theta*` с помощью защищенной (ограничим значения параметров следующими значениями `1e-3, 1-1e-3`) градиентной оптимизации $f(x + \text{AudioGenKS}(\theta))$. Не забудем поменять знак ошибки, чтобы добиться эффекта атаки. Таким образом, атаку можно считать grey-box атакой, так как у нас есть представление об архитектуре wake-word модели.
4. Полученные параметры `\theta` можно использовать для генерации аудио-атаки.

* параметры `\theta`: `bpm, freq, boost, transpose`

Результаты эксперимента:

1. Wake-word модель обучена до точности 95%
2. Обучен генератор струнной музыки на основе алгоритма Карплус-Стронг. При совмещении тестовой выборки с сгенерированной музыкой удалось понизить точность с 95% до 3%.