Устойчивая система голосовой активации для встраиваемого устройства с использованием глубокого обучения

Екатерина Чуйкова

Московский физико-технический институт
Факультет управления и прикладной математики
Кафедра интеллектуальных систем

Научный руководитель: к.физ-мат.н. П. Ю. Бойко

Голосовая активация (Keyword Spotting, Wake Word Detection)

Определение

Детекция с микрофона заранее заданной команды (wake word). Запускает выполнение определенных действий.

Дано

Запись с микрофона в режиме реального времени

Цель

Необходимо в режиме реального времени обнаружить заранее заданную фразу, если она была произнесена

Голосовая активация (Keyword Spotting, Wake Word Detection)

Применение

Используется для активации:

- Голосовых ассистентов (например, "Привет, Сири" , "Окей, Гугл")
- IoT устройств

Метрики

- Количество ложных срабатываний в час (False Accept Rate per hour)
- Процент ложных отклонений (False Reject Rate)

Цель работы

Задача

Обучить модель голосовой активации

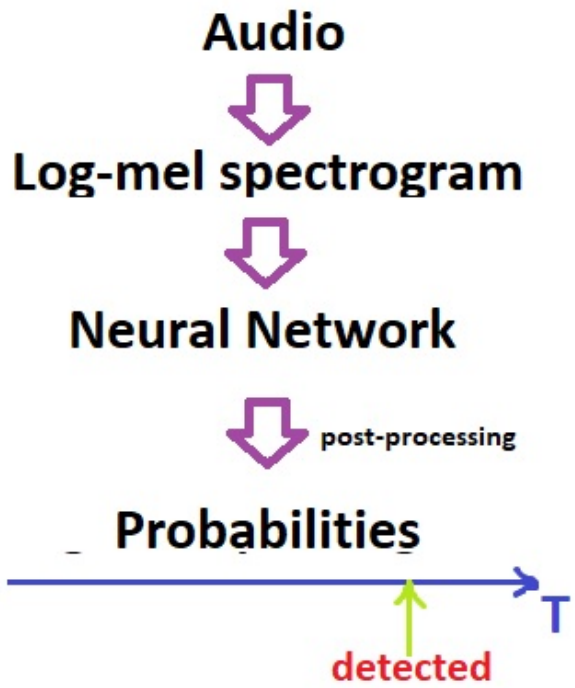
Требования к модели

- Низкий уровень ложных срабатываний (FA rate)
- Низкий уровень ложных отклонений (FR rate)
- Устойчивость к внешему шуму
- Low footprint (воспроизводимость на мобильном устройстве, минимальная нагрузка на cpu)
- Минимальная задержка при онлайн-детекции

Проблемы

Сложно добиться одновременно низких FA rate, FR rate, low footprint

Механизм работы



if p $>$ threshold $=>$ detected

Схема детекции wake word в поступающем аудио

Преобразование аудио в log-mel спектрограмму

Спектрограмма сигнала $s(t)$ может быть оценена путём вычисления квадрата амплитуды оконного преобразования Фурье сигнала $s(t)$, следующим образом, где w - некоторая оконная функция:

$$\text{spectrogram}(t, \omega) = |\,\text{STFT}(t, \omega)|^2$$

В данной работе используется окно Ханна (Хеннинга), где N — ширина окна

$$w(n) = 0.5\left(1 - \cos\left(\frac{2\pi n}{N-1}\right)\right)$$

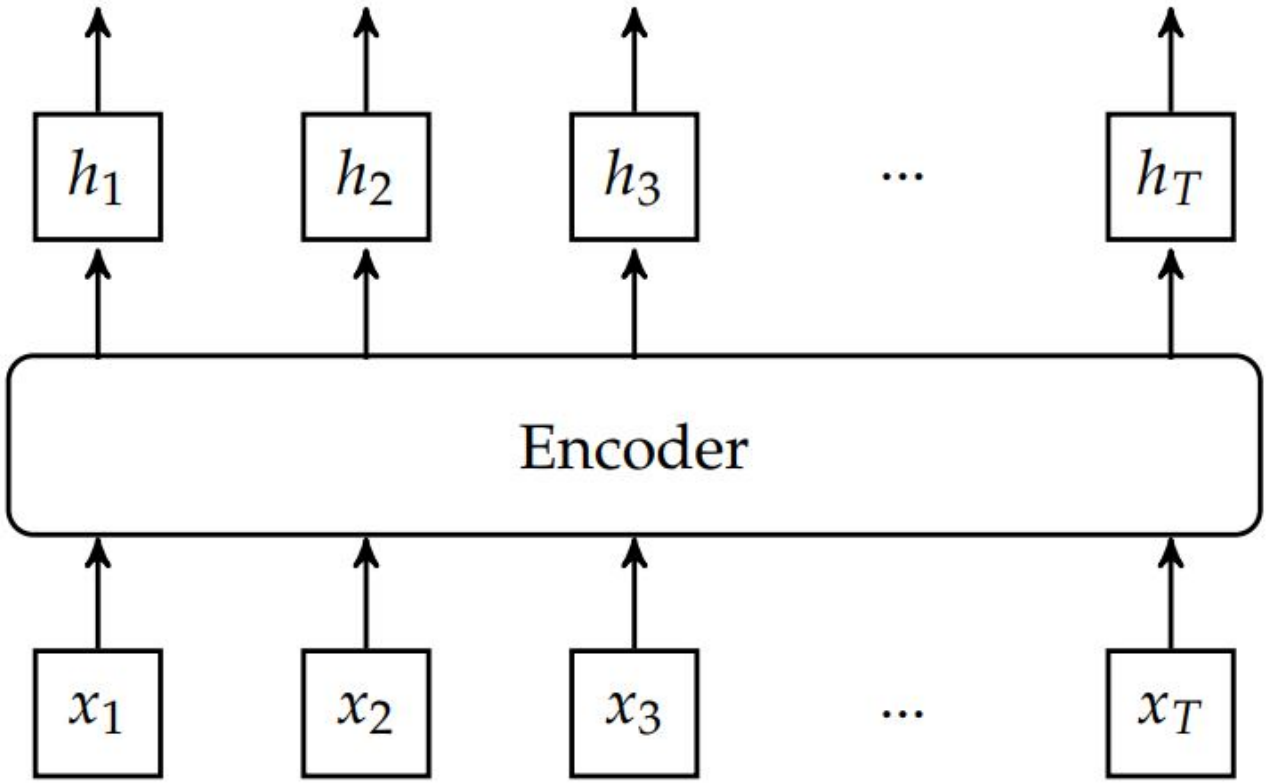
$$\text{melspectrogram}(t, \omega) = \text{spectrogram}(t, \omega) * \text{weightsmatrix}$$

$$\text{logmelspectrogram}(t, \omega) = \log(\text{melspectrogram}(t, \omega) + \epsilon)$$

Существующие исследования

- Small-footprint keyword spotting using deep neural networks (2014)
- Convolutional Recurrent Neural Networks for Small-Footprint Keyword Spotting (2017)
- Sequence-to-sequence Models for Small-Footprint Keyword Spotting (2018)

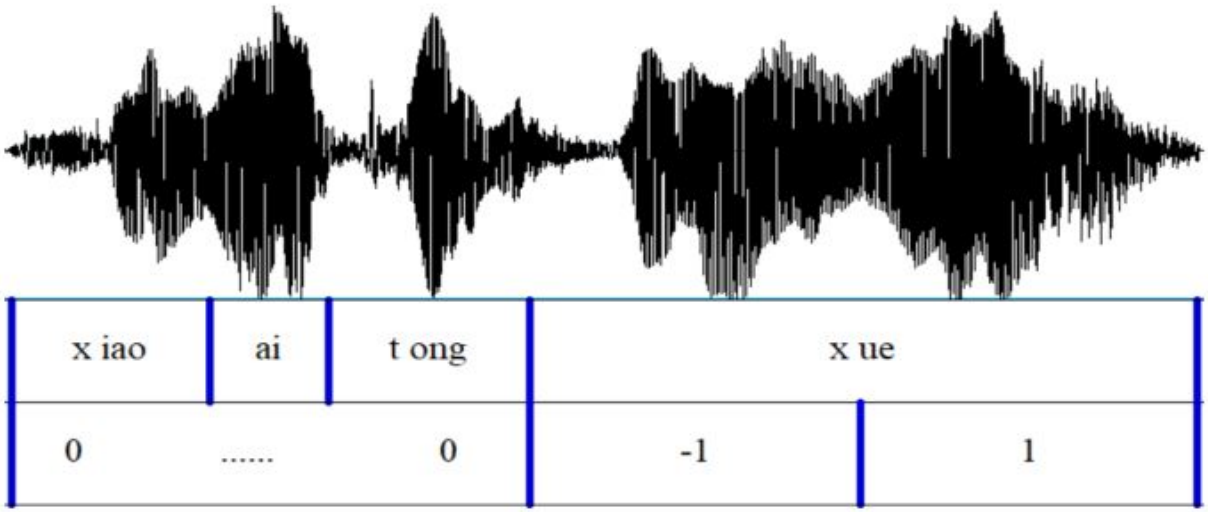
Архитектура сети



В качестве Encoder используется 1-2 слоя GRU, hidden size 64-256

Sequence-to-sequence Models for Small-Footprint Keyword Spotting, 2018

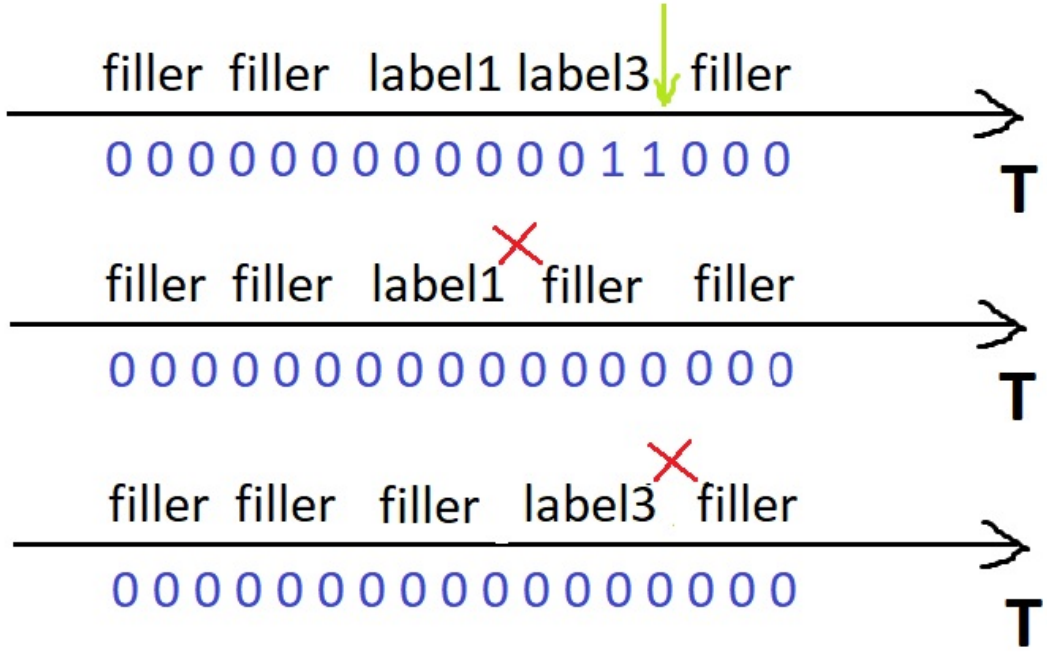
Разметка данных



Sequence-to-sequence Models for Small-Footprint Keyword Spotting, 2018

Разметка данных

wake word 1 = label1 + label3
wake word 2 = label2 + label3



Применение метода разметки данных из статьи к рассматриваемым данным

Разметка данных

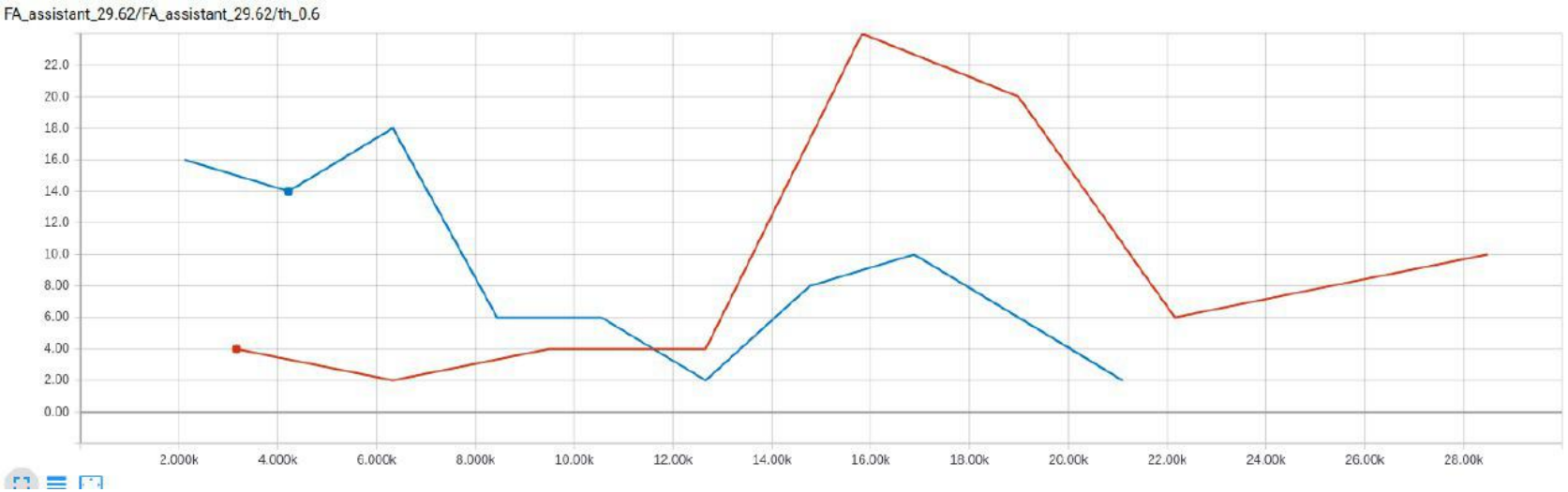
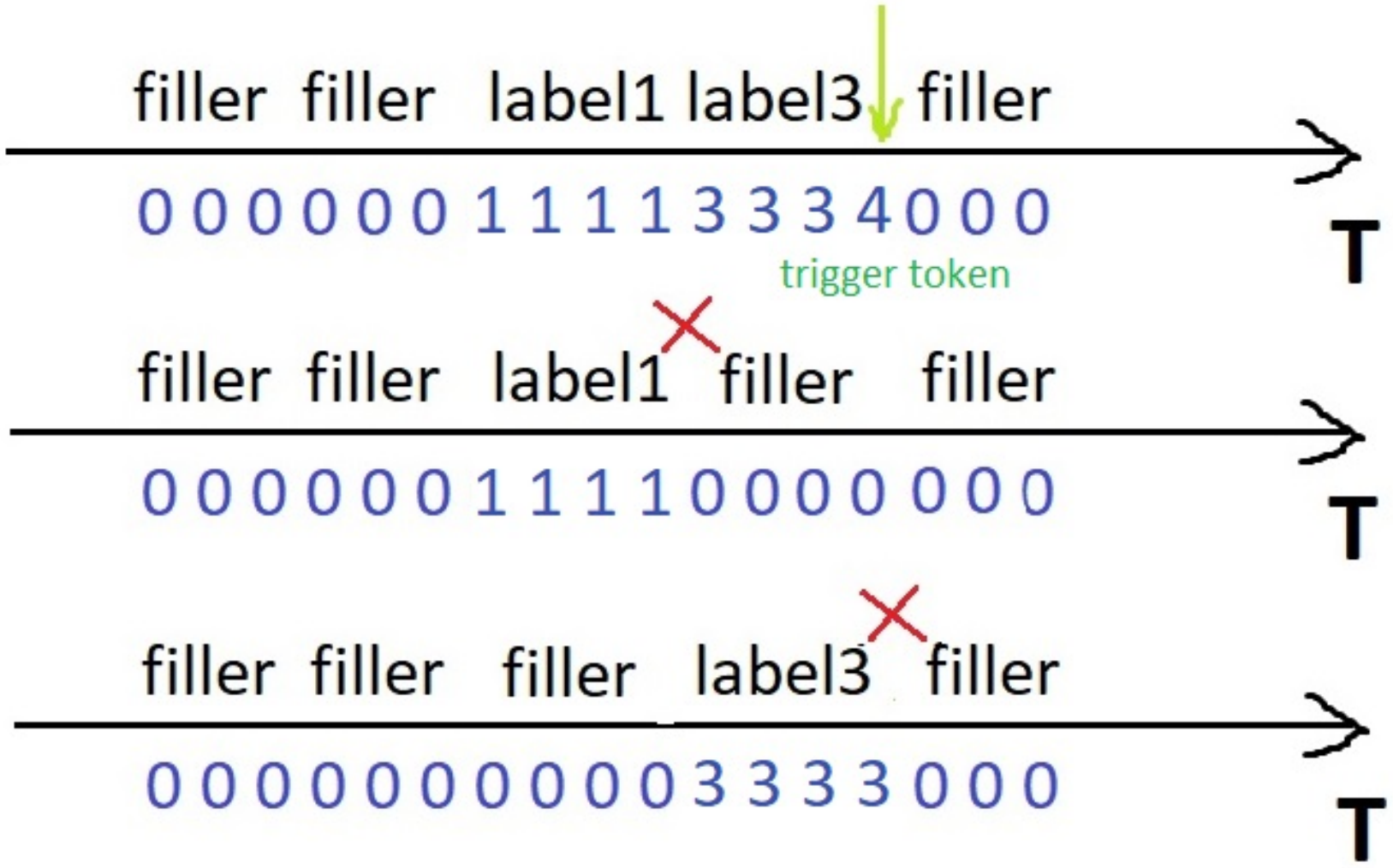


График FA rate при обучении
X - шаги обучения
Y - значение FA rate
(значения должны уменьшаться)

Разметка данных



Используемый метод разметки данных

Разметка данных

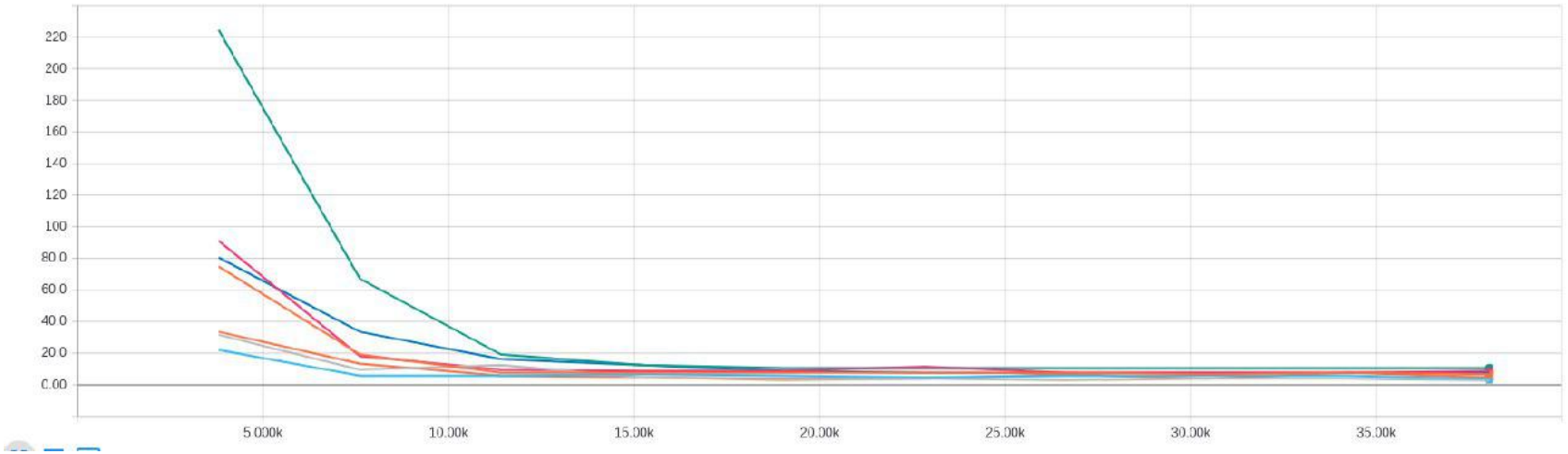
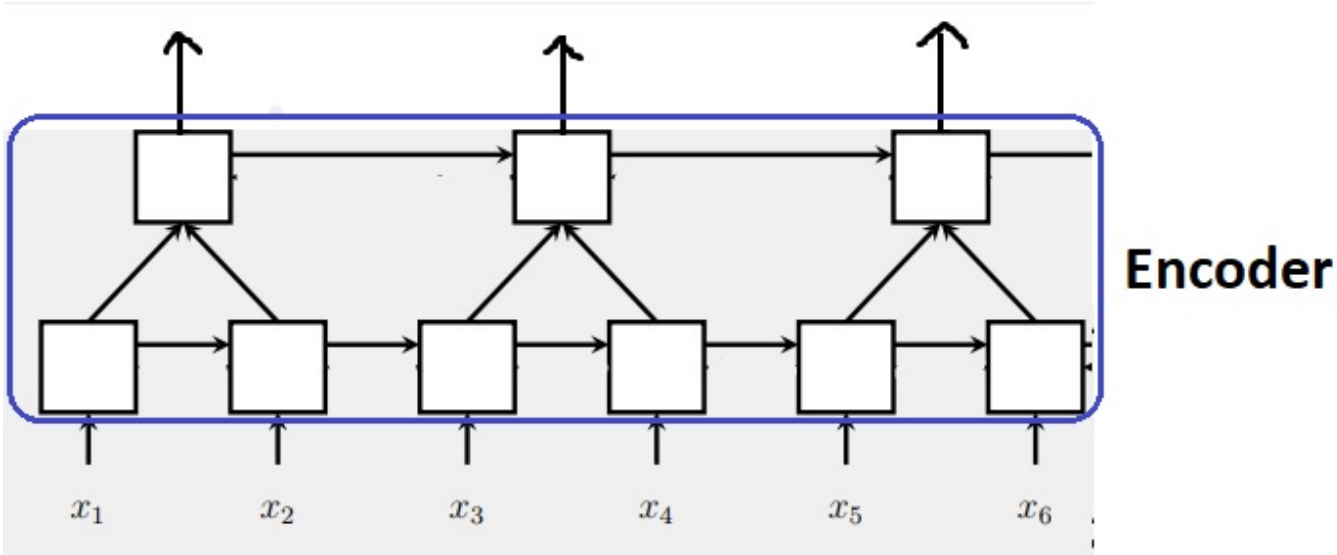


График FA rate при обучении
X - шаги обучения
Y - значение FA rate

Используемая архитектура сети



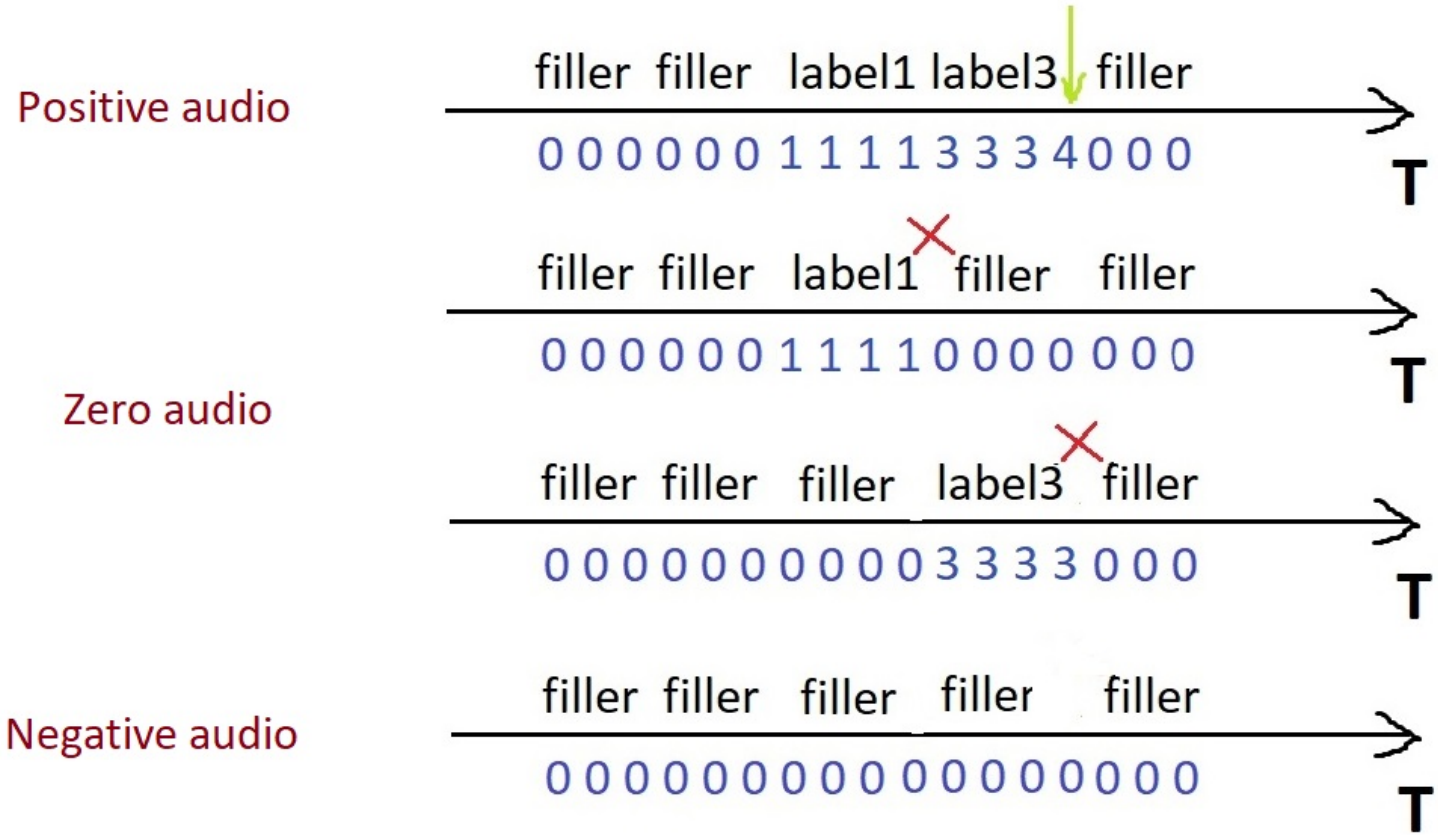
Pyramidal recurrent network*

*похожая архитектура (pBLSTM) использовалась в Listen, Attend and Spell, 2015

Init state для RNN

- Zero init state -> state degradation
- Zero init state, сброс state через некоторые промежутки времени при детекции
- Random init state
- Trainable init state
- State from samples distribution (еще не исследовался)

Данные



Типы данных для обучения

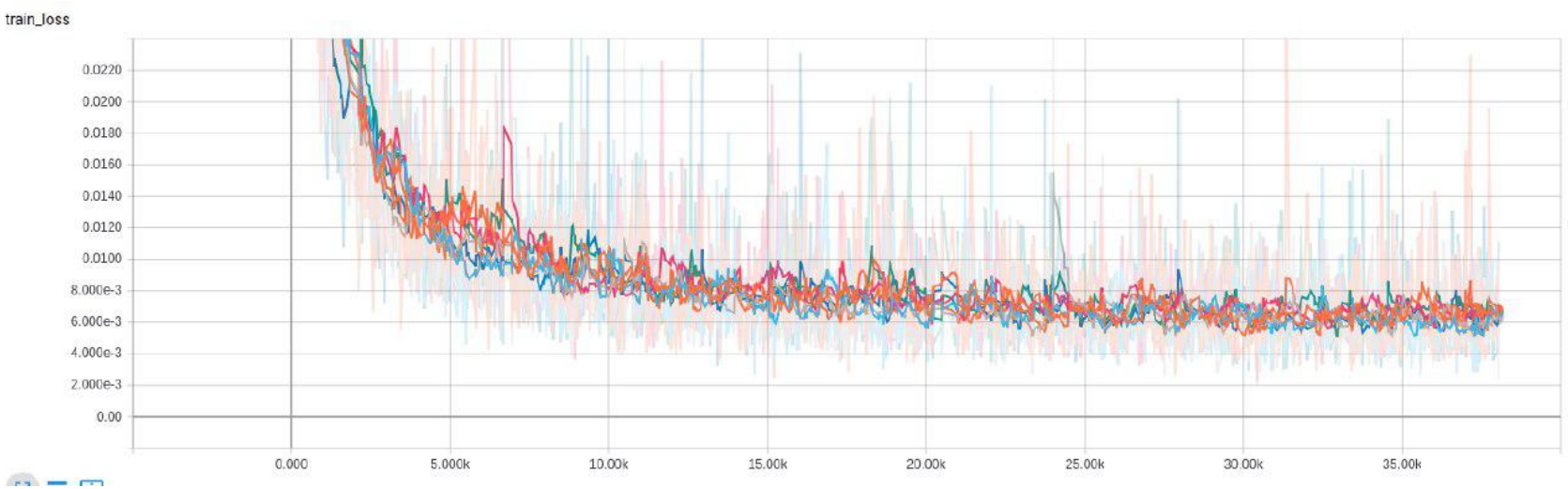
Данные

- 4 wake word, состящих из 2 комбинаций 4 слов
- 15.000 positive примеров для каждого wake word
- 1000 спикеров
- 135.000 zero примеров
- 400 часов negative примеров, включающих музыку, речь, тишину, шум.

Параметры обучения

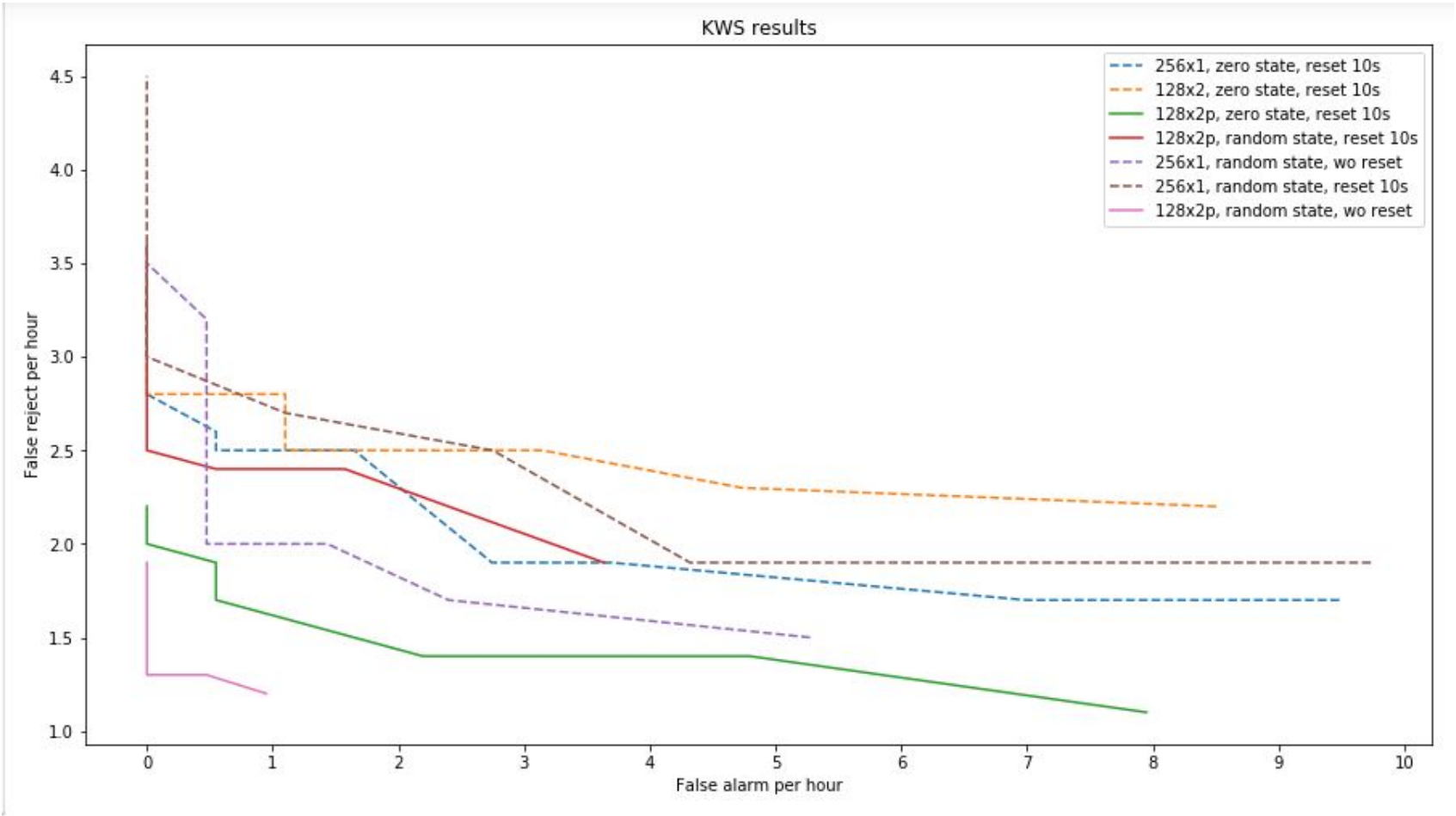
- batch size 80
- weight decay 1e-5
- start learning rate 2e-4, decay rate 0.9
- 10 epoch
- window size 0.02, window stride 0.01
- num mel bins 40
- GPU acceleration
- Tensorflow

Результаты



Train loss

Результаты



Полученные результаты. Лучше те результаты, графики которых лежат ниже

План исследования

- Исследовать семплирование init state, обучение из рандомных семплированных init state
- Исследовать аугментации аудио для повышения устойчивости к шуму
- Исследовать промежуточный способ разметки данных.