

# Interactive Speaker Recognition

Применение обучения с подкреплением для решения задачи  
распознавания диктора

Вячеслав Головин  
Евгений Шуранов (руководитель)

Huawei CBG AI и ФКН ВШЭ СПб

16.05.2023

# Задача распознавания диктора (Speaker Recognition)

Два типа задач:

- 1 **Идентификация** — по услышанной речи выбираем одного диктора из списка.
- 2 **Верификация** — по услышанной речи решаем, произнёс ли её конкретный диктор.

Фактически обе задачи сводятся к определению меры схожести между двумя наборами данных:

- 1 Векторы признаков, вычисленные из полученных ранее аудиозаписей речи (**эмбединги дикторов** или голосовые подписи).

Обозначение:  $G = [g^k]_{k=1}^K$ ,  $K \in \mathbb{N}$ .

- 2 Векторы признаков аудиозаписей речи, полученных сейчас (**эмбединги произнесенных слов**).

Обозначение:  $X = [x^t]_{t=1}^T$ ,  $T \in \mathbb{N}$ .

# Область исследования

## Зачем нам *Interactive Speaker Recognition*

Обычная постановка задачи speaker recognition:

- 1 Существует датасет с аудиозаписями речи различных дикторов.
- 2 Создаем модель идентификации / верификации.

В таких работах часто результатом становится модель, извлекающая качественные векторы признаков (эмбеддинги) из аудиозаписей речи.

Эта работа **не про эмбеддинги**. Здесь исследуется **задача выбора таких слов или фраз**, по которым уже существующая система будет распознавать диктора с более высокой точностью.

Исследуется метод, предложенный в статье *A Machine of Few Words — Interactive Speaker Recognition with Reinforcement Learning*, Mathieu Seurin et al., INTERSPEECH 2020, arXiv:2008.03127v1.

**Цель:** повышение точности систем верификации / идентификации диктора при помощи выбора запрашиваемых у диктора слов. Такая система может, например, быть использована для подтверждения личности на мобильных устройствах.

**Требования к системе:**

- короткие запросы (не раздражаем пользователя),
- разнообразные запросы (не боимся спуфинга),
- высокая точность (без комментариев).

**Предлагаемое решение:** использование RL-агента для выбора запрашиваемых слов.

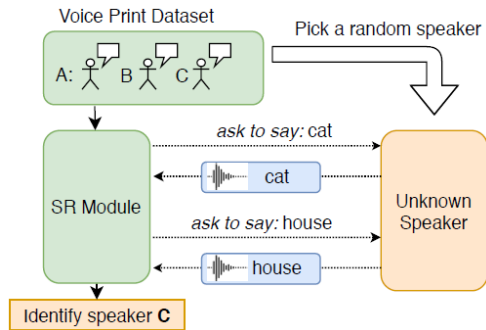
**Новизна дипломной работы:**

- переход от идентификации к верификации,
- более гибкая система для выбора слов,
- исследование влияния шума.

# Interactive Speaker Recognition

Здесь и далее изображения из *A Machine of Few Words — Interactive Speaker Recognition with Reinforcement Learning*, Mathieu Seurin et al., INTERSPEECH 2020, arXiv:2008.03127v1.

Использовался датасет TIMIT.

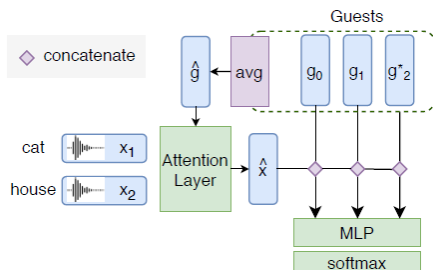


Важные особенности статьи:

- 1 только идентификация
- 2 фиксированный набор слов
- 3 разные нейронные сети для запроса слов (Enquirer) и идентификации (Guesser)

# Архитектура Guesser

Пытаемся угадать диктора



Входные данные:

- эмбединги дикторов  
 $G = [g_1; g_2; \dots g_K]$
- эмбединги слов  
 $X = [x_1; x_2; \dots x_T]$

Выходные данные:

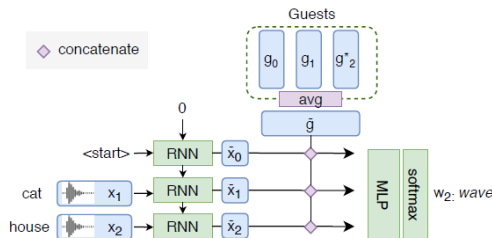
- вероятности  
 $\{P(g_i = g^*) \mid i = 1..K\}$

## Обозначения

- $K$  количество гостей / дикторов  
 $T$  количество запрашиваемых слов

# Архитектура Enquirer

Выбираем, какое слово мы спрашиваем у диктора



Входные данные:

- среднее эмб. дикторов  
$$\hat{g} = \frac{1}{K} \sum_{i=1}^K g_k$$
- эмбеддинги слов  
$$X = [x_1; x_2; \dots; x_t]$$

Выходные данные:

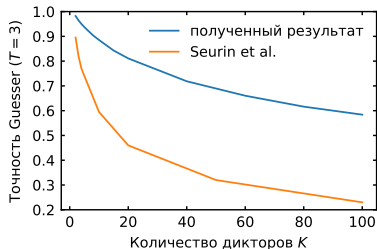
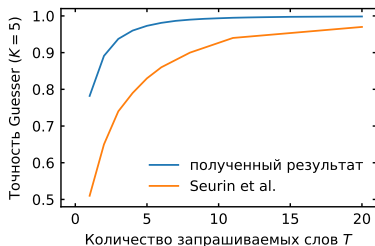
- вероятность выбрать  
каждое из слов

## Обозначения

- $K$  количество гостей / дикторов  
 $T$  количество запрашиваемых слов  
 $t$  количество запрошенных слов,  $0 \leq t \leq T$

# Обучение и тестирование Guesser

$K = 5$  дикторов и  $T = 3$  слова при обучении

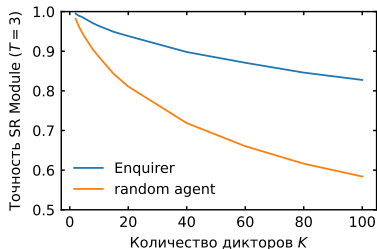
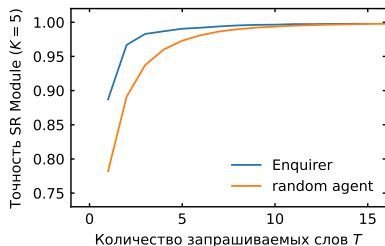


Вероятно, главная причина расхождения результатов — увеличение размерности эмбеддингов (512 вместо 128 в статье). Неизвестно, как и зачем в статье производилось понижение размерности.



# Обучение и тестирование Enquirer

$K = 5$  дикторов и  $T = 3$  слова при обучении

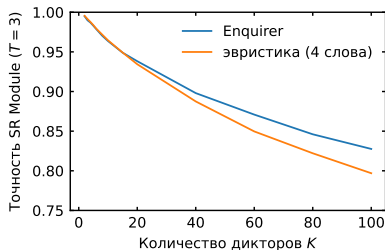
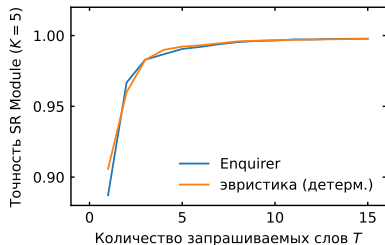


Для обучения использовалась **PPO**. Выбор слова при обучении и тестировании проводился по-разному:

- train — сэмплирование из распределения,
- test —  $\arg \max$  по не использованным ранее словам.

# Обучение и тестирование Enquirer

$K = 5$  дикторов и  $T = 3$  слова при обучении



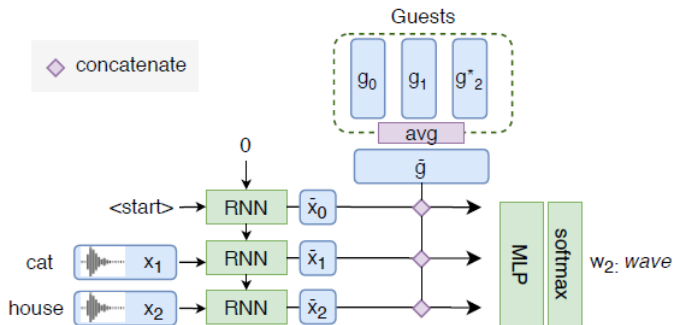
Для обучения использовалась **PPO**. Выбор слова при обучении и тестировании проводился по-разному:

- **train** — сэмплирование из распределения,
- **test** —  $\arg \max$  по не использованным ранее словам.

Эвристический агент не обращает внимание на контекст и (практически) всегда запрашивает одни и те же слова.

# От идентификации к верификации

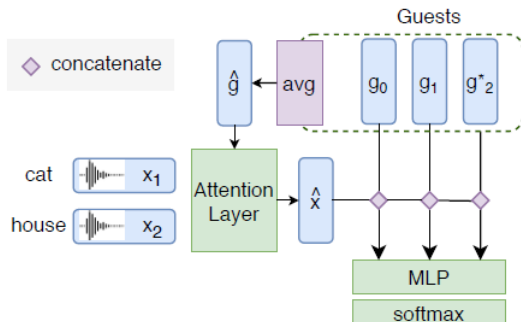
$T = 3$  слова



- Enquirer: не меняем ничего (даже веса)

# От идентификации к верификации

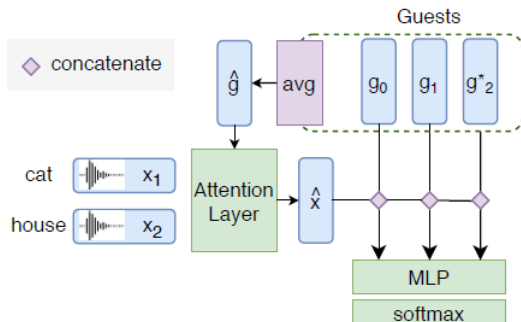
$T = 3$  слова



- Enquirer: не меняем ничего (даже веса)
- Guesser: меняем softmax на sigmoid

# От идентификации к верификации

$T = 3$  слова



- Enquirer: не меняем ничего (даже веса)
- Guesser: меняем softmax на sigmoid

Выбор слов	Точность
случайный	0.895
Enquirer	0.933
эвристика	0.917

## Обучение в более тяжелом режиме

Выбор слов	Режим обучения	Точность
случайный	$T = 3$	0.895
Enquirer		0.933
эвристика		0.917
случайный	$T = 2$	0.913
Enquirer		0.947
эвристика		0.945

Таблица: Точность верификации,  $T = 3$  запрашиваемых слова

## Обучение в более тяжелом режиме

Выбор слов	Режим обучения	Точность
случайный	$K = 5$ $T = 3$	0.937
Enquirer		0.982
эвристика		0.984
случайный	$K = 20$ $T = 2$	0.951
Enquirer		0.989
эвристика		0.988

**Таблица:** Точность идентификации,  $K = 5$  дикторов,  $T = 3$  запрашиваемых слова

## Другие эксперименты

### ❶ CodebookEnquirer — гибкая система выбора слов.

- ▶ Меняем последний слой Enquirer: теперь он выдает не вероятности выбрать то или иное слово из словаря, а эмбеddинг.
- ▶ Создаем Codebook — набор эмбеddингов слов (усредняем по обучающей выборке).
- ▶ Для получения вероятностей считаем softmax с отрицательной температурой от расстояний между выходным эмбеddингом и эмбеddингами слов в Codebook.
- ▶ Работает (небольшое падение точности), даже если мы обучаем и тестируем модель на разных наборах слов.

### ❷ Добавление шума

- ▶ Добавляем к аудиозаписям слов 6 видов шума из MUSAN.
- ▶ Не меняем тип шума в течение игры.
- ▶ Не помогает Enquirer опережать эвристику.



# Выводы

- Исследованный подход работает — точность идентификации существенно повышается при добавлении выбирающего слова агента.
- Модель можно сделать практически полезной: легко перейти от идентификации к верификации и от фиксированного набора слов к произвольному.
- В большинстве режимов (очень) простая эвристика оказывается не хуже нейросетевого агента для выбора слов (Enquirer).