Московский государственный университет имени М.В.Ломоносова

Механико-математический факультет

Кафедра Математической теории интеллектуальных систем и лабораторий

Киназаров Темирбек

«Исследование свойств голоса с помощью рекуррентных нейронных сетей»

Курсовая работа

Научный руководитель:

доцет по кафедре Математической теории интеллектуальных систем и лабораторий,

кандидат физико-математических наук

Часовских А. А.

Москва, 2019

Оглавление

Введение

В данной курсовой работе проводится анализ работы [2] и проведения экспериментов распознования дикторов по голосовым сообщениям и анализ параметров для точности распознавания.

Основные понятия, связанные с нейронными сетями, как об автоматах, были взяты из фундаментальных трудов по теории автоматов Кудрявцева В.Б., Алешина С.В., Подколзина А.С. *"Введение в теорию автоматов"*.

В работе были рассмотрены методы, описанные в исследованиях [2]. В этой работе в исследовании данного вопроса было выяснено и замечено, что кроме методов классического машинного обучения с этой задачей отлично справиться может и LSTM. В статье Андрея Карпатого [6] подробно описан данный вид рекурентных нейронных сетей. Эта работа показывает хорошую работоспособность метода LSTM в последовательности данных, что характерно для звуковых дорожек. Эта теория отлично описана в статье разработчиков из Google [2].

Были проаналлизированы описанные методы и модели, были проведены эксперименты, анесены некторые изменения в работу данной модели, показавшие достаточно высокие результаты в распознавании и отделении векторов голосовых дорожек отдельных дикторов на явные признаковые поля. Для обучения использовались базы данных на основе англоговорящих дикторов, но модель хорошо показала себя и на базе данных, основанной на русскоговорящих дикторах, что дает возможность исследования в сторону малой или отсутствующей зависимости от языка, на котором были произнесены фразы дикторами.

Постановка задачи

Дается размеченная база данных, состоящей из файлов формата .wav, которые содержат записи дикторов, произносящих различные фразы необязательно одинаковые.

Нужно на основе этой базы данных научиться определять с некоторой точностью, кому пренадлежит новая запись, которая, возможно, не хранится в данной базе.

Требуется также попытка реализовать отсутствие зависимости модели от содержания фраз, то есть большое внимание характеристикам голоса.

Метод решения

При решении было рассмотрено большое количество методов и подходов для обработки звука [2] [3] [4]. Для решения данной задачи нужно на основе размеченной базы данных, разработать метод для создания векторов, подкрепленых общей логикой за каждой фразой отдельных дикторов.

В первую очередь в работе рассматриваются методы для обработки звука переводящие их в вектора признаков, содержащие в себе достаточное количество информации, используемые для распознавания речи. Данную часть можно назвать "предобработкой".

Предобработка

В работе был использован классический метод обработки аудиофайлов, содержащие речевые дорожки, а точнее метод мел-кепстральных коэффициентов (MFCC) [7].

Рис. 1: Схема действий в предобработке

Очередность действий, использованных для создания вектора признаков данным методом обработки, изображена на схеме(рис. 1).

В системах распознавания по голосу данный метод считается одним из самых распространенных. Суть метода заключается в следующем:

1. Подача последовательности результата сигментации по тактам $\chi_0, \chi_1, ..., \chi_{N-1}$.

2. Применение весовой функции для уменьшения искажений. Чаще всего в качестве весовой функции используют окно Хэмминга:

$$\omega_n = 0,54 - 0,46 \cos\left(2\pi \frac{n}{N-1}\right), n = 0, ..., N-1,$$

где N - размер окна в отсчетах. Коэффициенты получены империческим путем, их можно менять, что, возможно, даст другие результаты.

3. Применение дискретное преобразование Фурье (ДПФ):

$$X_k = \Sigma_{n=0}^{N-1} \chi_n \omega_n e^{\frac{-2\pi i}{N} kn}, k = 0, ..., N-1.$$

где k соответстует частотам $f_k = \frac{F_s}{N} k, k = 0, ..., N/2$, где F_s является частотой дискритизации.

4. Далее с помощью треугольных фильтров идет разбиение на диапазоны. Границы этих фильтров рассчитываются в шкале мел. Мел - единица высоты звука, основанная на восприятии этого звука нашим слухом. Формула для перевода в мел-частотную область:

$$B(f) = 1127 * \ln\left(1 + \frac{f}{100}\right)$$

Формула обратного преобразования:

$$B^{-1}(b) = 700(e^{b/1127} - 1)$$

Чаще всего используют 24 фильтра [9](стр. 2). Количество фильтров обозначим как N_{FB}. Фильтры применяются к квадратам модулей коэффициентов преобразования Фурье, а затем высчитывается логарифм:

$$e_m = \ln\left(\Sigma_{m=0}^{N} \mid X_k \mid^2 H_{m,k}\right), m = 0, ..., N_{FB} - 1$$

где $H_{m,k}$ - весовые коэффициенты фильтров, которые были получены.

5. Дискретное косинусное преобразование является последним этапом данного метода. На этой стадии происходит вычисление мел-частотных кепстральных коэффициентов (MFCC):

$$c_i = \Sigma_{m=0}^{N_{FB}} e_m \cos\left(\frac{\pi i(m + 0,5)}{N_{FB}}\right), i = 1, ..., N_{MFCC}$$

Коэффициент c_0 - энергия сигнала, поэтому он не используется. В нашем случае

количество мел-частотных кепстральных коэффициентов равняется порядка 12.

Полученные вектора данным методом используются в распозновании речи, что означает содержание информации о сказанных фразах, поэтому сделанной работы до данного момента было недостаточно.

Пример в найденом исследовании(рис.2) [10], изображена модель, где синими отмечены мужские голоса и красными - женские, которая илюстрирует смешанность и плохую отделимость данных голосов.

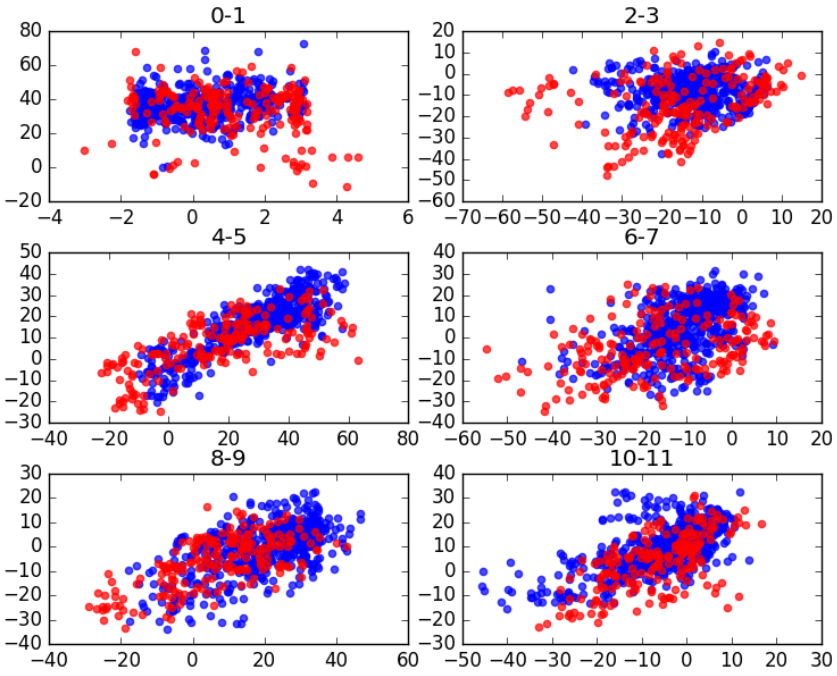

Рис. 2: Пример плохой разделимости признакового пространства построенных в методе предобработки

Применение LSTM и функция ощибки

Полученные вектора признаков сильно зависят от сказанных фраз каждого диктора. Необходимо использовать функцию ошибки такую, чтобы избавиться от этих свойств.

Ниже приведена схема дальнейших действий в рис.3.

Следующее действие заключается в подаче в сеть LSTM предобработанных данных в виде отмеченных последовательностей.

В работе обучена сеть LSTM с тремя слоями по 512 узлов со входным вектором размера 128 элементов.

Для того, чтобы обучить сеть необходимо выбрать функцию ошибки.

В работе рассматривается метод triplet loss или Triplet Similarity Embedding, который использовался в статьях моделей Facenet от Google [8] и "Deep face recognition"(VGG). Идея данной функции: приблизить как можно ближе вектора, полученные на выходе, одного класса к их среднему вектору (центроиде) путем коррекции параметров слоев.

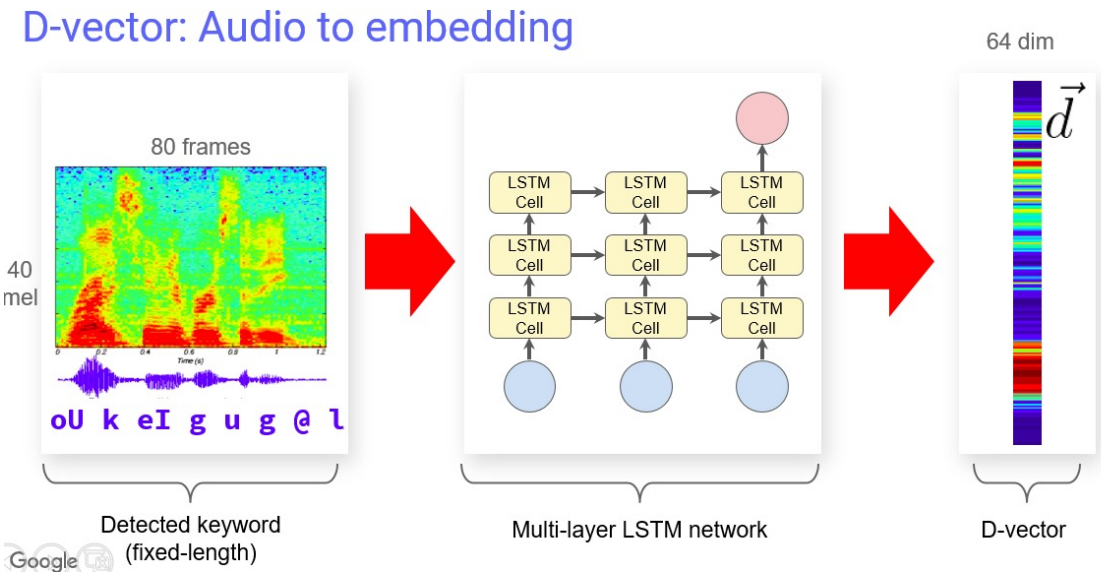

Рис. 3: Схема действий в модели

Но также для повышения точности была использована модификация метода(End-To-End), которая не только собирала в кучи элементы одного класса, но и отдаляла их от средних векторов других классов.

Если говорить подробнее о методе, то после применения LSTM получались вектора $f(x_{ji}; w)$ фиксированной длины, отмеченные под каждого диктора на основе прогоняемых данных, где x_{ji} - полученные последовательности от j-ого диктора произнесенной i-ой фразы, а w - условно обозначенные веса. Далее для уменьшения разброса их стоит нормализовать, то есть получить:

$$e_{ji} = \frac{f(x_{ji}; w)}{\|f(x_{ji}; w)\|}$$

Средние вектора (или центроиды) ищутся по формуле

$$c_k = \frac{1}{M}\Sigma_{i=1}^{M} e_{ki}$$

Далее составляется матрица из трех индексов

$$S_{ji,k} = w * \cos(e_{ji}, c_k) + b$$

называемая в статье Similarity Matrix.

Как можно заметить $S_{ji,k}$ зависит от значения угла наклона каждого векора от каждой центроиды как своего класса, так и остальных. То есть каждый элемент матрицы хранит в себе информацию отдаленности векторов от центроид, что мы можем использовать для корректировки весов.

Функция ошибки каждого вектора высчитывается, как

$$L(e_{ji}) = -S_{ji,j} + \log \Sigma_{k=1}^{N} \exp(S_{ji,k})$$

Финальная функция ошибки получается суммированием этих значений по всем векторам:

$$L_G(x; w) = L_G(S) = \Sigma_{j,i} L(e_{ji})$$

Численный эксперимент

Для проверки работоспособности на выбранных датасетах достаточно было использовать готовый метод классификации. В курсовой был выбран метод опорных векторов (SVM) [11]. И проверкой являлось попадание выборки в полученные поля.

Для обучения модели использовался датасет "TIMIT"`https://catalog.ldc.upenn.edu/LDC93S1`. После этого полученные коэффициенты были проверены на датасете из англоговрящих дикторов на одинаковых фразах без поторонних шумов`https://datashare.is.ed.ac.uk/bitstream/handle/10283/1942/clean_trainset_wav.zip`. И на русскоговорящих дикторах с разнородными фразами `http://www.repository.voxforge1.org/downloads/Russian/Trunk/Audio/Original/16kHz_16bit/`, где неожиданно были показаны результаты сравнимые с предыдущими. Также объектом исследования было уменьшение датасета путем уменьшения количества файлов для каждого диктора.

	Кол-во дикторов	Кол-во фраз	Правильные ответы,
"Чистый"датасет англ.	29	350-400	79
Зашумленый датасет рус.	10	200	80-85
Малый зашумленый датасет	10	12	79-95

Разработанная программа с подробной инструкцией по запуску и вложенная презентация по курсовой работе находятся по адресу `https://github.com/TimurKinazar/coursework`.

Заключение

В данной работе был проведен эксперимент по верификации дикторов. Результаты эксперимента показывают, что нейронная сеть справляется с точностью не меньшей, чем 79 процентов.

К сожалению точность распознавания не получается более 80 процентов в среднем из всех проведенных опытов. Все приведенные выше действия привели к точности только в диапозоне 79 - 85 процентов, а другие более высокие результаты являются случайными, возможно, в результате выборки из датасета для проверки.

В перспективе имеется идея для увеличения точности добавления дополнительных нелинейных слоев сети в промежутке между результатом обработки LSTM и функции ошибки. В дальнейшем можно применить другие методы классификации в последнем шаге нашего эксперимента вместо SVM.

Отдельную благодарность хочется выразить Анатолию Александровичу Часовских за помощь в проведении работы и предоставлению задачи, а также Половникову Владимиру Сергеевичу за предоставленное оборудоваение для проведение экспериментов и поправки. Также хотелось бы выразить отдельную благодарность Ронжину Дмитрию Владимировичу за руководство в процессе исследования и проведения экспериментов.

Литература

1. Кудрявцев В.Б., Алешин С.В., Подколзин А.С. *"Введение в теорию автоматов*
2. Li Wan, Quan Wang, Alan Papir, Ignacio Lopez Moreno *"Generalized End-to-End Loss for Speaker Verification"* — https://arxiv.org/abs/1710.10467
3. Yun Tang, Guohong Ding, Jing Huang, Xiaodong He, Bowen Zhou *"Deep Speaker Embedding Learning with Multi-Level Pooling for Text-Independent Speaker Verification"* — https://arxiv.org/abs/1902.07821
4. Hagen Soltau, Michael Picheny, David Nahamoo,George Saon *"Speaker adaptation of neural network acoustic models using i-vectors"* — https://www.researchgate.net/publication/261485126_Speaker_adaptation_of_neural_network_acoustic_models_using_i-vectors
5. Harry Volek *"PyTorch implementation of "Generalized End-to-End Loss for Speaker Verification"by Wan, Li et al"* — https://github.com/HarryVolek/PyTorch_Speaker_Verification
6. Andrej Karpathy *"Understanding LSTM Networks"* — https://colah.github.io/posts/2015-08-Understanding-LSTMs
7. Иванов И.И. *"Анализ метода мел-частотных кепстарльных коэффициентов применительно к процедуре голосовой аутентификации"* — Журнал: Актуальные проблемы гуманитарных и естественных наук. 2015 г. Номер: 10-1. стр.106-114
8. Florian Schroff, Dmitry Kalenichenko, James Philbin *"FaceNet: A Unified Embedding for Face Recognition and Clustering"* — https://arxiv.org/abs/1503.03832
9. Ронкер В.Ю. *"Автоматическое распознавание человека в диалоговых системах"* — conf.sfu-kras.ru/sites/mn2013/thesis/s044/s044-042.pdf
10. Станислав Рыжов *"Анализ мел-частотных кепстральных коэффициентов (MFCC)"* — http://ai-for-everybody.blogspot.com/2016/11/mfcc.html
11. Сергей Николенко *"SVM и kernel methods* — https://logic.pdmi.ras.ru/~sergey/teaching/mlau12/08-svm.pdf