Математические структуры и моделирование
2023. № 4 (68). С. 93–99

УДК 004.032.26
DOI 10.24147/2222-8772.2023.4.93-99

МОДЕЛЬ РАСПОЗНАВАНИЯ ПОЛА ДИКТОРА НА ОСНОВЕ АУДИОДАННЫХ

А.Т. Мухаматханова
студент, e-mail: mukhamatkhanovaalsu@gmail.com
Т.М. Опарина
старший преподаватель, e-mail: oparina2007@yandex.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. Рассматривается задача распознавания пола диктора по аудиоданным с использованием метода k-ближайших соседей (KNN). Авторы предлагают алгоритм извлечения признаков из спектрограмм аудиозаписей, основанный на вычислении центроида, ширины спектра и спада высоких частот. Затем полученные признаки используются в качестве входных данных для метода KNN. Результаты показывают, что предложенный метод позволяет достичь высокой точности распознавания пола диктора.

Ключевые слова: машинное обучение, анализ аудиоданных.

Введение

В современном мире разработка систем идентификации пола диктора становится все более актуальной. Такие системы находят широкое применение в различных областях, включая криминалистику и колл-центры. Данная работа представляет собой исследование и реализацию алгоритма машинного обучения, предназначенного для автоматического распознавания мужского и женского голосов на основе анализа аудиоданных. Для реализации алгоритма будет использоваться язык программирования Python, а также специализированные библиотеки, такие как Scikit-Learn, Pandas, NumPy, Matplotlib и другие. Любые аудиоданные можно представить как функцию от амплитуды и времени. Для понимания машиной аудиоданных они должны быть представлены в виде числовых данных. Для этого необходимо из аудиоданных извлечь акустические характеристики. Будем использовать библиотеку Librosa – пакет Python для анализа звука [1, 2]. В качестве источника данных для обучения и тестирования алгоритма будет использоваться датасет аудиоданных «Common Voice».

1. Работа с набором данных. Анализ данных

Для начала проанализируем данные датасета. Используем метод read_csv() для чтения данных и функцию head() для вывода. Датасет содержит 8 колонок с данными (см. рис. 1).

```
data=pd.read_csv('common-voice/cv-valid-train.csv')
data.head(5)
```

	filename	text	up_votes	down_votes	age	gender	accent	duration
0	cv-valid-train/sample-000000.mp3	learn to recognize omens and follow them the o...	1	0	NaN	NaN	NaN	NaN
1	cv-valid-train/sample-000001.mp3	everything in the universe evolved he said	1	0	NaN	NaN	NaN	NaN
2	cv-valid-train/sample-000002.mp3	you came so that you could learn about your dr...	1	0	NaN	NaN	NaN	NaN
3	cv-valid-train/sample-000003.mp3	so now i fear nothing because it was those ome...	1	0	NaN	NaN	NaN	NaN
4	cv-valid-train/sample-000004.mp3	if you start your emails with greetings let me...	3	2	NaN	NaN	NaN	NaN

Рис. 1. Содержимое датасета

Исходный датасет включает много пустых данных и данные, не нужные для распознавания мужского и женского голоса. Однако данные датасета являются реалистичными, включая различные аудиодорожки с людьми из разных частей мира и с определённым акцентом. Для решения задачи будет достаточно только самого аудиофайла и пола человека, который произносит текст в нём. Из датасета мы выбираем только два столбца – «filename» и «gender». С помощью функции notna() проверяем данные на наличие пустых значений и формируем новый датасет (см. рис. 2).

```
data = data[['filename','gender']]
new_data = data[data['gender'].notna()]
new_data.reset_index(inplace=True, drop=True)
new_data.head()
```

	filename	gender
0	cv-valid-train/sample-000005.mp3	female
1	cv-valid-train/sample-000008.mp3	male
2	cv-valid-train/sample-000013.mp3	female
3	cv-valid-train/sample-000014.mp3	male
4	cv-valid-train/sample-000019.mp3	male

Рис. 2. Датасет после удаления ненужных данных

Значение столбца «gender» важно для классификации. Для распознавания нужно, чтобы классификация была бинарная, следовательно, значений в этом столбце должно быть два – «мужчина» и «женщина». Но в столбце «gender», помимо двух значений, которые необходимы, присутствует ещё и третье значение. Это данные, в которых люди, произносящие текст, решили не оглашать свой пол. Следующим шагом необходимо удалить данные, у которых значение столбца «gender» равняется «other». Проведя анализ данных и преобразования, получили датасет, который содержит в себе только необходимые данные для поставленной задачи.

2. Извлечение признаков из аудиодорожек

Извлечение признаков является важным этапом в аудиоанализе. Отберём признаки, которые являются ключевыми в определении пола человека. Для этого используем функции, с помощью которых будем извлекать данные признаки [3]:

1. feature_spectral_centroid – функция для определения спектрального центроида. Указывает, на какой частоте сосредоточена энергия спектра или, другими словами, где расположен «центр масс» для звука;

2. feature_spectral_bandwidth – функция для определения спектральной ширины;

3. feature_spectral_rolloff – функция для определения спектрального спада. Это мера формы сигнала, представляющая собой частоту, в которой высокие частоты снижаются до 0;

4. feature_mfcc_mean – мел-кепстральные коэффициенты. Представляют собой набор признаков, которые описывают общую форму спектральной огибающей. Они моделируют характеристики человеческого голоса.

Датасет с извлечёнными данными содержит в себе 23 столбца с числовыми данными, которые являются важными для обучения, и один столбец с названием пола (см. рис. 3). Данные являются числовыми, они уже понятны для машины, но имеют очень большие значения и отклонения. Для дальнейшей работы и правильной классификации их нужно привести к стандартному виду.

	gender	1	2	3	4	5	6	7	8	9	...	14	15	16
19787	0	2676.649813	2665.705767	5318.154351	-531.11615	106.982056	-1.045521	28.467484	10.845261	4.290282	...	0.313017	-4.977850	-6.315069
19788	1	2814.714591	3405.077399	5438.105194	-548.00793	104.883804	53.127365	34.703370	12.573208	22.028343	...	4.388365	-4.981477	2.755410
19789	0	2408.574034	2368.667478	4312.204817	-432.23120	138.888120	13.700328	32.105200	-1.449688	9.888707	...	0.639174	-8.937414	-0.169536
19790	0	1885.397634	1950.331935	3298.793039	-606.87350	129.088350	22.133833	23.852484	14.771211	-2.254710	...	-5.886711	-11.928243	-0.183971
19791	0	3071.184919	3171.170628	6326.787961	-424.43845	124.160520	-11.763888	23.819141	1.263500	8.397840	...	-8.623758	-14.255218	-0.836812

5 rows × 24 columns

Рис. 3. Датасет с извлеченными данными

3. Подготовка данных к обучению

Следующим этапом необходимо подготовить данные к обучению. Для этого делаем следующие шаги:

1. Разделяем данные на два вектора. Вектор «Y» – целевой вектор. Он содержит в себе значения, которые нужно предугадать, т. е. значение столбца «gender». А вектор «X» – вектор признаков, содержит в себе все остальные данные, которые важны для обучения.
   Y=df_features.gender.values
   X=df_features.drop(["gender],axis=1).

2. Стандартизация данных. Стандартизация – процесс преобразования исходного набора данных в новый со средним значением – 0 и стандартным отклонением – 1. Стандартизация позволяет устранить возможное влияние отклонений по какому-либо признаку. Признаки датасета часто имеют большие различия в своих диапазонах, поэтому стандартизация необходима. А в алгоритмах, которые вычисляют расстояние между точками на графике, отсутствие стандартизации может привести к неправильному восприятию данных. Стандартизацию будем выполнять с помощью функции StandardScaler().

3. Разделение данных. Разделение данных на обучающую и тестовую выборки необходимо для оценки способности модели к обобщению на новых данных, которых не было в обучающей выборке. Если модель будет обучена на данных из обучающей выборки и затем протестирована на тех же данных, она может показать высокую точность предсказаний, но при этом плохо справиться с новыми данными. Обучать модель будем на данных «X_train» и «Y_train», а делать прогнозы и проверять качество модели – на «X_test» и «Y_test». Разделение проводится с помощью функции train_test_split из библиотеки Scikit-learn. Для обучения возьмём 80 % данных, а для тестирования – 20 %.

4. Метод k-ближайших соседей (KNN)

При реализации данного алгоритма используются следующие методы:

1. Метод init – это функция инициализации переменных;

2. Метод fit – функция, которая сохраняет тренировочный набор данных;

3. Метод predict – функция предсказания меток класса;

4. Метод find_labels – функция расчёта расстояния по каждому тестовому наблюдению для каждого наблюдения из тренировочного набора;

5. Метод distance – функция расчёта расстояния. Для расчёта расстояния используем Евклидово расстояние [4]:

$$D = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}.$$

6. Метод «most_common» – функция определения наиболее частого класса;

7. Метод «score» – функция для определения точности алгоритма;

8. Для алгоритма очень важно правильно подобрать параметр «k» – число соседей. Значения «k» от 2 до 10 с шагом 1. Далее на графике можно отследить точность алгоритма в зависимости от значения «k» (см. рис. 4). На графике видно, что наивысшая точность достигается при $k = 2$ и постепенно уменьшается с увеличением этого параметра.

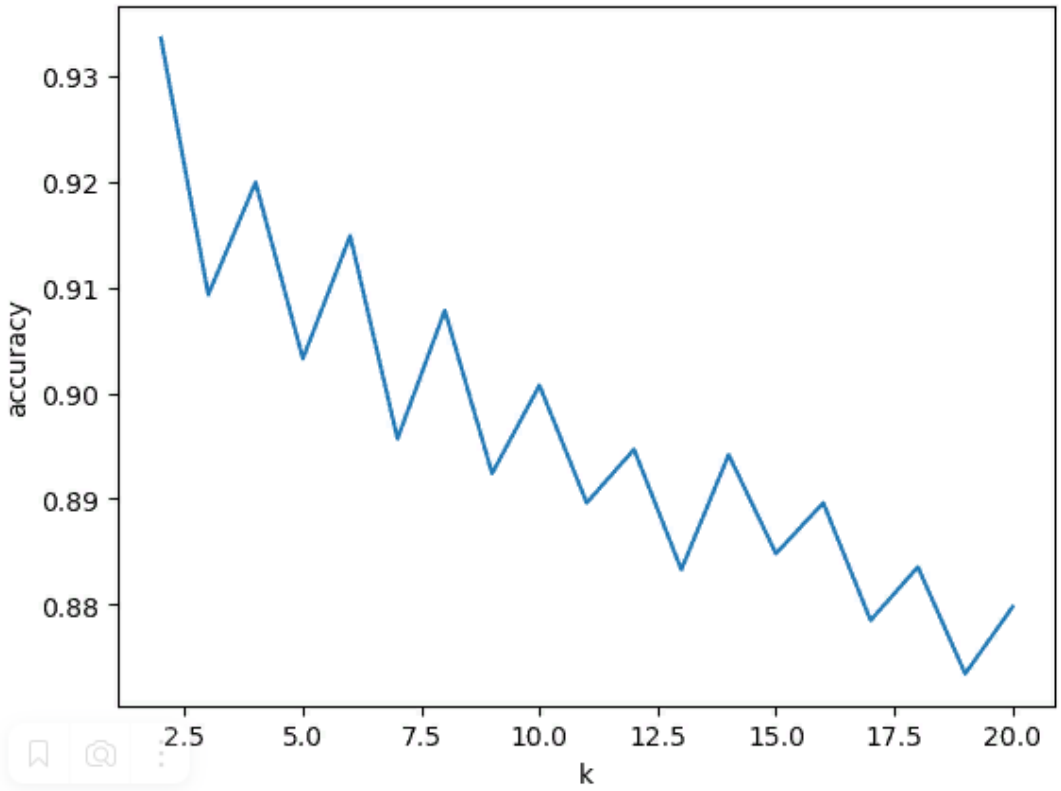

Рис. 4. График зависимости точности алгоритма от параметра k

5. Анализ результатов

Точность алгоритма составила 88 %. Из 1004 мужчин алгоритм предугадал 990, а ошибся только на 14, а при предугадывании женщин он сделал 219 ошибок. Таким образом, алгоритм лучше распознает мужчин, чем женщин, такая проблема может возникнуть при несбалансированности данных. В нашем датасете много данных, но количество мужчин сильно превосходит количество женщин. Посмотрим количество данных по каждому классу, из 15833 данных мужчины составляют 11871, а женщины – 3962. С несбалансированными данными можно справиться с помощью функции RandomUnderSampler() – случайная недостаточная выборка. Её работа заключается в том, что она случайным образом уменьшает количество классов большинства до желаемого соотношения, по сравнению с классом меньшинства. Данный метод подходит к нашему датасету, потому что мы имеем достаточно большую выборку, в которой существует много экземпляров, расположенных близко друг к другу (см. рис. 5). После преобразования данных количество классов с меткой 0 стало равным количеству классов с меткой 1. Оно уменьшилось до 3962. Теперь алгоритм будет обучаться на 7924 данных. Этого количества данных достаточно, чтобы алгоритм хорошо обучился. На сбалансированных данных точность алгоритма составила 92 %, что на 4 % больше, чем при обучении с несбалансированными данными. Обученная машина теперь может распознавать мужчин и женщин.

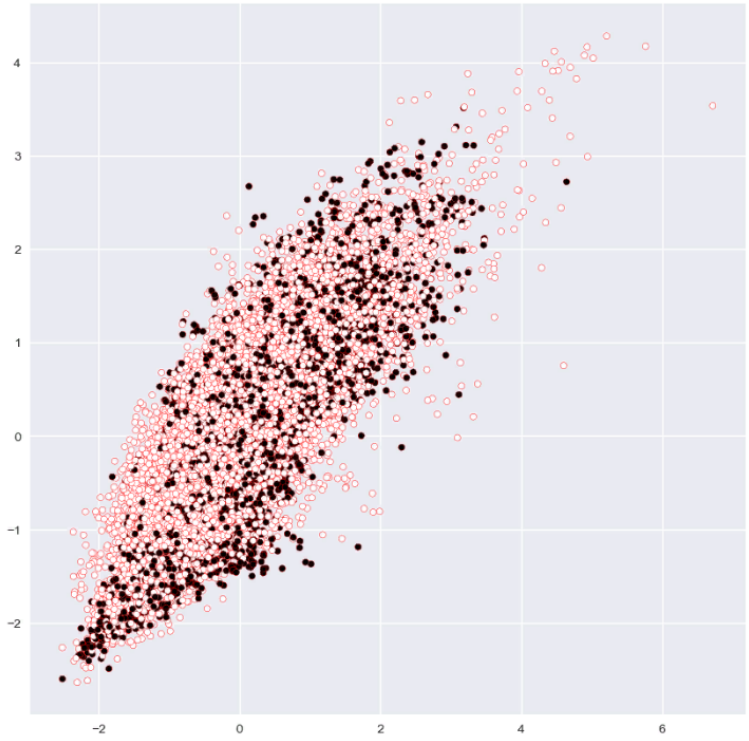

Рис. 5. График данных датасета

Для проверки результата использовали датасет, который содержал 100 записей: 40 мужчин и 60 женщин. Эти данные не были включены в тренировочную и тестовую выборки. В итоге правильно были определены 40 мужчин (ошибся только на 6 женщинах). В итоге получили, что данный алгоритм машинного обучения распознает мужские и женские голоса из аудиоданных на 94 %.

Литература

1. Руководство Librosa 0.10.1. URL: `https://librosa.org/doc/latest/tutorial.html` (дата обращения: 20.11.2023).
2. Вандер П.Дж. Python для сложных задач: наука о данных и машинное обучение. СПб. : Питер, 2018. 576 с.
3. Черкасов А.Н., Грибко И.И. Разработка системы распознавания естественного языка для идентификации голосовых данных // Вестник АГУ. 2021. Вып. 4 (291), № 3. С. 75–80.
4. Кугаевских А.В., Муромцев Д.И., Кирсанова О.В. Классические методы машинного обучения. СПб. : Университет ИТМО, 2022. 53 с.

MODEL OF SPEAKER GENDER RECOGNITION BASED ON AUDIO DATA

A.T. Mukhamatkhanova
Student, e-mail: mukhamatkhanovaalsu@gmail.com
T.M. Oparina
Assistant Professor, e-mail: oparina2007@yandex.ru

Dostoevsky Omsk State University, Omsk, Russia

Abstract. In this article considered the task of recognizing the speaker's gender based on audio data using the KNN method. The authors propose an algorithm for extracting features from spectrogram of audio recordings based on calculating the centroid, bandwidth, and decay of high frequencies. The extracted features are then used as input data for the KNN method. The results show that the proposed method allows achieving high accuracy in recognizing the gender of the speaker.

Keywords: machine learning, audio data analysis.

Дата поступления в редакцию: 24.11.2023