به نام خدا

دانشکده سامانه‌های هوشمند

پردازش گفتار (۱۴۸-۰۵-۸۳)

نیم‌سال دوم ۱۴۰۳-۱۴۰۲

تمرین شماره ۴

تاریخ تحویل: ۱۴۰۳/۰۳/۰۹

---

۱. (۱۰٪) [**پژوهش**] تاکنون پروژه‌های مختلفی در حوزه یادگیری عمیق برای پردازش گفتار پیاده‌سازی و ارائه شده است (مثلا Tacatron و Vall-E برای تبدیل متن به گفتار و Wav2Vec و Whisper برای تبدیل گفتار به متن). گزارشی از مطالعه این پروژه‌ها برای سه حوزه بازشناسی گفتار، تبدیل متن به گفتار و بهسازی گفتار ارائه دهید که در آن برای هر حوزه حداقل دو پروژه را پوشش دهد. در گزارش خود، علاوه بر بیان مختصر مشخصات آن پروژه، مشخص کنید هر کدام از آنها کدامیک از الگوریتم‌های یادگیری ماشین را دارا استفاده کرده‌اند و برای هر حوزه، مقایسه مختصری از پروژه‌ها با همدیگر ارائه دهید.

---

۲. (۵۰٪)[**پیاده‌سازی: تشخیص اعداد با شبکه عصبی پرسپترون چندلایه (MLP)**]

از شبکه عصبی پرسپترون چندلایه برای تشخیص اعداد ۰ تا ۹ انگلیسی بهره ببرید. برای این کار از دادگان AudioMNIST (https://www.kaggle.com/datasets/sripaadsrinivasan/audio-mnist) بهره ببرید. در این دادگان، ۳۰۰۰۰ رکورد در ۱۰ کلاس اعداد ۰ تا ۹ انگلیسی وجود دارد که ۶۰ گوینده گوناگون آنها را گفته اند. شبکه MLP باید توسط خودتان پیاده‌سازی گردد و نباید از کتابخانه‌های آماده بهره گرفته شود. برای این پرسش:

گام ۱- فراخوانی دادگان و بخش‌بندی ۲۰ / ۸۰ آن: این داده، ۶۰ پوشه دارد که برای هر گوینده است و در هر پوشه، شماری فایل صدا است که گوینده آن پوشه آنها را گفته است. لازم است که تمامی ۳۰۰۰۰ صدا را در کنار هم داشته باشید و سپس ۲۰ / ۸۰ را اعمال کنید.

گام ۲- پیش‌پردازش‌های لازم روی داده: از روش MFCC با طول فریم ۲۵ میلی ثانیه، ۲۴ فلیتر مل و ۱۲ ویژگی با مشتق‌های مرتبه یک و دو بهره ببرید.

گام ۳- ساخت مدل MLP: از آنجایی که باید ۱۰ کلاس را دسته بندی کنیم، تعداد ۱۰ نرون باید در خروجی شبکه عصبی قرارداده شود. شمار نرون‌های ورودی شبکه نیز باید به اندازه ویژگی‌های گرفته شده از هر فایل صدا باشد (اتصال بردارهای فریم‌های متوالی)

- یک لایه میانی (مخفی) برای شبکه قرار دهید. برای شمار نرون‌های لایه میانی، میانگین شمارگان نرو ن‌های لایه ورودی و خروجی درنظر بگیرید.
- تابع فعالسازی لایه‌های میانی و خروجی، سیگموید دوقطبی باشد. مقدار نرخ یادگیری را برابر با ۰.۰۰۱ قرار دهید.

h.veisi@ut.ac.ir

- تابع هدف برای بهینه شدن را MSE در نظر بگیرید و از رویکرد (Online updating) SGDدر آموزش بهره ببرید.
- در پایان، از softmax برای یافت بهترین کلاس بهره ببرید.

گام ۴- آموزش و آزمون مدل:

الف- شبکه را با الگوریتم پس‌انتشار خطا آموزش دهید. برای کار یک اندازه کردن همه فایلها با هم، می‌توانید تا از روش zero padding بهره ببرید. مقدار Accuracy را برای هر دسته و به صورت میانگین و نمودار خطا MSE را برحسب تکرارها گزارش کنید. برای ثابت نگه داشتن تعداد نرون‌های ورودی، می‌توانید از میانگین گرفتن از بردارهای ویژگی فریم‌ها یک فایل میانگین بگیرید و نتیجه شبکه را با روش قبلی مقایسه کنید. چه راهی دیگری برای این کار می‌توانید پیشنهاد دهید؟

ب- تعداد لایه‌های میانی را به دو لایه افزایش دهید و بخش آ را در حالت‌های زیر تکرار کنید:

- شمارگان نرون‌ها در هر لایه میانی برابر با آنچه در حالت تک لایه بوده، درنظر بگیرید.
- شمارگان نرون‌های لایه یکم میانی، دو سوم و نیم شمارگان نرون‌های لایه ورودی باشد.
- شمارگان نرون‌های لایه یکم میانی، نیم و یک سوم شمارگان نرون‌های لایه ورودی باشد.

پ- از تابع فعال‌ساز ReLU- برای انجام بندهای الف و ب بهره ببرید. نتایج‌ها را باهم مقایسه کنید.

۳. (۴۰٪) [پیاده‌سازی: تشخیص ژانر موسیقی با شبکه عصبی پیچشی(CNN)]

در این تمرین بنا داریم تا ژانرهای موسیقیایی را با طراحی شبکه‌های عصبی پیچشی از هم تشخیص دهیم. دادگان این پرسش، GTZAN (https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification) است که ۱۰ کلاس ژانر موسیقی در آن است. همه این صداها، طول ۳۰ ثانیه دارد. برای پیاده‌سازی شبکه CNN می‌توانید از کتابخانه‌ها و ابزارهای دلخواه بهره ببرید.

گام ۱- فراخوانی دادگان و بخش‌بندی ۲۰ / ۸۰ آن: این داده، ۱۰ پوشه دارد که هر کدام بیانگر یک ژانر (کلاس) است و در هر پوشه، ۱۰۰ فایل صدا است. لازم است که تمامی صداها را در کنار هم داشته باشید و سپس ۲۰ / ۸۰ را اعمال کنید.

گام ۲- ورودی شبکه: از دادگان، طیف‌نگار (Spectrogram) هر فایل را استخراج کنید (لزومی به هم طول کردن فایل‌ها نیست) و برای ورودی شبکه برگزینید.

گام ۳- ساخت و آموزش مدل: بعد از لایه ورودی، دو لایه میانی (مخفی) پیچش و سپس یک لایه flatten بگذارید. ساختار همه لایه‌های پیچش، به صورت زیر است:

- یک لایه پیچش دو بعدی با تعداد ۳۲ کانال و اندازه کرنل ۳ * ۳ و تابع فعالسازی relu
- سپس یک لایه max pooling با اندازه کرنل ۳ * ۳ و stride برابر با ۲ * ۲ و padding مناسب
- در پایان، یک لایه batch normalization بگذارید.

- در لایه خروجی، یک لایه dense بگذارید. از آنجایی که باید ۱۰ کلاس را دسته‌بندی کنیم، تعداد ۱۰ نرون باید در خروجی شبکه عصبی قرارداده شود.
- در آموزش مدل، از بهینه‌ساز adam بهره ببرید. مقدار نرخ یادگیری را برابر با ۰.۰۰۰۱ قرار دهید. همچنین batch size را برابر با ۳۲ و تعداد epoch را برابر ۳۰ در نظر بگیرید.
- تابع هدف برای بهینه شدن را sparse categorical cross entropy در نظر بگیرید.

گام۴- آزمون مدل: همچون پرسش قبلی، Accuracy را برای این پرسش نیز گزارش کنید.

۴. (۲۰٪ امتیازی) [پیاده‌سازی: تشخیص گوینده]

فرض کنید شما به عنوان یک فرد متخصص داده کاوی در یک شرکت استخدام شده‌اید. وظیفه شما تشخیص گفتار پنج رئیس‌جمهور با نام های Benjamin Netanyahu, Jens Stoltenberg, Julia Gillard, Margaret Tacher, Nelson Mandela است. مسئله دسته‌بندی صداها بیشتر با ویژگی‌هایی از سیگنال گفتار بنام MFCC انجام می‌گیرد. در اینجا بجای MFCC، از سه ویژگی Energy، Zero Crossing Rate و Spectral centroid بهره ببریم. برای بدست آوردن این ویژگی‌ها، کتابخانه librosa می‌تواند کمک‌کننده باشد (در مورد این سه ویژگی، سر کلاس TA، صحبت خواهد شد). حال بنا داریم تا با بهره‌گیری از الگوریتم‌هایی که در درس یادگیری ماشین داشتیم کار دسته‌بندی را انجام دهیم.

الف) یکی از مهم ترین مراحل یادگیری ماشین جمع‌آوری داده است. فرض کنید از شما خواسته شده است که یک مجموعه داده(dataset) از هر کدام از این پنج فرد تهیه کنید. تحقیق کنید برای اینکه مدل‌های ما، داده‌های لازم و کافی برای آموزش داشته باشند، به ازای هر کلاس(تشخیص هر فرد)، معمولا پیشنهاد می‌شود چه میزان داده، با چه طولی و با چه ویژگی‌هایی جمع آوری شود. فرض کنید داده‌های

h.veisi@ut.ac.ir

به نام خدا

دانشکده سامانه‌های هوشمند

پردازش گفتار (۱۴۸-۰۵-۸۳)

نیم‌سال دوم ۱۴۰۳-۱۴۰۲

تمرین شماره ۴

تاریخ تحویل: ۱۴۰۳/۰۳/۰۹

مورد نیاز را جمع آوری کردیم و در لینک زیر به آن‌ها دسترسی داریم (Each audio in the folder is a one-second 16000 sample rate PCM encoded).

https://www.kaggle.com/datasets/kongaevans/speaker-recognition-dataset/data

در بین پوشه ها فقط به پنج پوشه‌ی متعلق به پنج فرد کار داریم. بقیه‌ی پوشه را نادیده بگیرید. در این بخش، می‌خواهیم تا با بکارگیری مدل Logistic Regression، کار دسته‌بندی را انجام دهید. برای این بخش باید:

گام ۱- دادگان خوانده شود.

گام ۲- تابعی برای استخراج این سه ویژگی از فایل‌های صدا بسازید و دادگان را فراهم کنید (برای ورودی مدل این ویژگی‌ها را به صورت نرمالایز شده (یکی از روش‌ها کفایت می‌کند) به مدل بدهید)

- https://www.geeksforgeeks.org/what-is-data-normalization/
- https://librosa.org/doc/main/generated/librosa.feature.zero_crossing_rate.html
- https://librosa.org/doc/main/generated/librosa.feature.rms.html
- https://librosa.org/doc/main/generated/librosa.feature.spectral_centroid.html

گام ۳- میزان زمان لازم (executed time) برای استخراج سه ویژگی از کل داده ها را در حالت عادی به دست آورید. حال از آن جایی که هر فایل صدا به صورت جداگانه در حال استخراج ویژگی‌ها هستند و وابستگی ندارند، پس برای استخراج ویژگی‌ها می‌توان از برنامه‌نویسی موازی روی سی‌پی‌یو کمک گرفت برای این منظور پیشنهاد می‌شود از کتابخانه‌ی پایین استفاده کنید.

```
from concurrent.futures import ThreadPoolExecutor
```

حال زمان لازم برای بدست اوردن ویژگی‌های کل داده‌ها در این حالت را بدست اورید و با حالت قبل مقایسه کنید.

- https://pynative.com/python-get-execution-time-of-program/
- https://superfastpython.com/threadpoolexecutor-in-python/

گام۴- به صورت تصادفی، ۸۰٪ از دادگان برای آموزش و ۲۰٪ باقی‌مانده برای آزمون مدل تقسیم شود.

h.veisi@ut.ac.ir

دقت کنید که مدل در آموزش خود نباید از این ۲۰٪ داده را ببیند.

گام ۵- مدل Logistic Regression را با بهره‌گیری از کتابخانه‌هایی که این مدل را دارند، مانند sklearn، با رویکرد گرادیان کاهشی، آموزش دهید. دقت کل، ماتریس درهم‌ریختگی (confusion matrix) و معیارهای F1 score، Recall و Precession را گزارش دهید. در این آموزش، تعداد iteration را ۱۰۰۰ و نرخ یادگیری را ۰.۰۱ بدانید.

ب) برای کار دسته‌بندی پیشنهاد شده، بجای بهره‌گیری از سه ویژگی یادشده، از MFCC بهره ببرید. MFCC و MFCC_mean را می‌توانید با قطعه کد زیر بدست آورید و نتایج را گزارش کنید.

```
n_mfcc = 13
MFCC = librosa.feature.mfcc(signal, n_fft= number_of_samples_per_fft,
                            hop_length=shift_value, n_mfcc= n_mfcc)
MFCC_mean = np.maen(mfcc, axis=1)
```

پ) در برنامه‌ای که شما نوشته‌اید احتمالا هر صدایی بدهیم به یکی از این پنج کلاس پیش بینی می‌شود (مثلا اگر صدای یک بز را بدهیم می‌گوید این صدا مربوط به کلاس Nelson Mandela است در صورتی که نیست و باید پیش‌بینی شود که به هیچ کدام از این پنج دسته متعلق نیست) برای این که واقع‌گرایانه‌تر باشد صرفا تحقیق کنید که برای رفع این مشکل داده‌کاوان چه روش‌هایی استفاده می‌کنند.

h.veisi@ut.ac.ir

به نام خدا

دانشکده سامانه‌های هوشمند

پردازش گفتار (۱۴۸-۰۵-۸۳)

نیم‌سال دوم ۱۴۰۳-۱۴۰۲

تمرین شماره ۴

تاریخ تحویل: ۱۴۰۳/۰۳/۰۹

h.veisi@ut.ac.ir