تشخیص احساسات گفتاری

فهرست مطالب

معرفی

اهمیت تشخیص احساسات با بهبود تجربه کاربر و تعامل رابط های کاربری صوتی (VUI) رواج پیدا می کند. توسعه سیستم های تشخیص احساسات مبتنی بر گفتار دارای مزایای کاربردی عملی است. با این حال ، این مزایا تا حدی نادیده گرفته می شود که سر و صدای پس زمینه واقعی باعث اختلال در عملکرد تشخیص احساسات مبتنی بر گفتار هنگام استفاده از سیستم در برنامه های کاربردی عملی می شود.
تشخیص احساسات گفتاری (SER) یکی از چالش برانگیزترین کارها در حوزه تجزیه و تحلیل سیگنال گفتار است ، این یک مشکل حوزه تحقیقاتی است که سعی می کند احساسات را از سیگنال های گفتاری استنباط کند.

در واقع تشخیص احساسات فرایند شناسایی احساسات انسانی است. دقت افراد در تشخیص احساسات دیگران بسیار متفاوت است. استفاده از فناوری برای کمک به افراد در تشخیص احساسات یک حوزه تحقیقاتی نسبتاً نوپا است. به طور کلی ، این فناوری در صورتی که از چند روش در زمینه استفاده کند ، بهترین عملکرد را دارد. تا به امروز ، بیشترین کار بر روی خودکار تشخیص حالات چهره از طریق ویدئو ، عبارات گفتاری از طریق صدا ، عبارات نوشتاری از متن و فیزیولوژی است که توسط پوشیدنی ها اندازه گیری می شود.

تشخیص احساسات یکی از مهمترین استراتژی های بازاریابی در دنیای امروز است. شما می توانید موارد مختلف را برای یک فرد به طور خاص متناسب با علاقه خود شخصی سازی کنید.به همین دلیل ، ما تصمیم گرفتیم پروژه ای را انجام دهیم که بتوانیم احساسات افراد را فقط با صدای آنها تشخیص دهیم که به ما اجازه می دهد بسیاری از برنامه های مرتبط با هوش مصنوعی را مدیریت کنیم. برخی از مثالها می

تواند شامل مراکز تماس برای پخش موسیقی هنگام عصبانی شدن فرد در تماس باشد. یکی دیگر می تواند یک ماشین هوشمند باشد که هنگام عصبانیت یا ترس سرعت خود را کاهش می دهد.در نتیجه این نوع برنامه دارای پتانسیل زیادی در جهان است که می تواند به نفع شرکت ها و حتی ایمنی مصرف کنندگان باشد.

رابط کاربری صوتی (VUI) تعامل گفتاری انسان با رایانه ها را ممکن می سازد ، از تشخیص گفتار برای درک دستورات گفتاری و پاسخ به سوالات استفاده می کند و معمولاً برای پخش پاسخ از متن به گفتار استفاده می کند. دستگاه فرمان صوتی (VCD) دستگاهی است که با رابط کاربری صوتی کنترل می شود.

SER اگرچه چندان محبوب نیست ، SER در این سالها حوزه های زیادی را وارد کرده است ، از جمله:

حوزه پزشکی: در دنیای پزشکی از راه دور که بیماران در بسترهای تلفن همراه مورد ارزیابی قرار می گیرند ، توانایی یک متخصص پزشکی در تشخیص احساس بیمار در واقع می تواند در روند بهبود مفید باشد.

خدمات به مشتریان: در مرکز تماس از مکالمه برای تجزیه و تحلیل مطالعه رفتاری همراهان تماس با مشتریان استفاده می شود که به بهبود کیفیت خدمات کمک می کند.

سیستم های توصیه گر: توصیه می شود محصولات را بر اساس احساسات مشتریان نسبت به آن محصول به مشتریان توصیه کنید.

متدولوژی

اول ، ما باید برخی از وابستگی ها را با استفاده از pip نصب کنیم:

Required Dependencies

- **Librosa**
- **Numpy**
- **Scikit-learn**
- **keras**

کل روند به شرح زیر است (مانند هر روند یادگیری ماشین):

آماده سازی مجموعه داده: در اینجا ، ما مجموعه داده را بارگیری و تبدیل می کنیم تا برای استخراج مناسب باشد.

بارگیری مجموعه داده: این فرایند در مورد بارگیری مجموعه داده در پایتون است که شامل استخراج ویژگی های صوتی است ، مانند بدست آوردن ویژگی های مختلف مانند قدرت ، صدای پیکربندی مجرای صوتی از سیگنال گفتار ، ما از کتابخانه librosa برای این کار استفاده می کنیم.

آموزش مدل: پس از آماده سازی و بارگیری مجموعه داده ، ما آن را به سادگی بر روی یک مدل sklearn مناسب آموزش می دهیم.

آزمایش مدل: اندازه گیری عملکرد و دقت در مدل.

ابتدا ، ما به یک مجموعه داده برای آموزش نیاز داریم ، خوشبختانه مجموعه داده RAVDESS در آدرس https://smartlaboratory.org/ravdess/ وجود دارد ، من آن را از لینک زیر

https://zenodo.org/record/1188976#.XN0fwnUzZhE

بارگیری کرده و با موفقیت استخراج کردم. پس از آن ما باید میزان نمونه را در تمام فایل های صوتی کاهش دهیم تا librosa تکمیل گردد.

اجازه دهید تابعی را ایجاد کنیم که ویژگی های استخراج را مدیریت می کند (که شکل موج گفتار را به شکل نمایش پارامتری با سرعت نسبتاً کمتر داده تغییر می دهد):

سازماندهی فایل های صوتی

مرحله بعدی شامل سازماندهی فایل های صوتی است. هر فایل صوتی دارای یک شناسه منحصر به فرد در موقعیت ششم نام فایل است که می تواند برای تعیین احساسات فایل صوتی استفاده شود. ما در مجموعه داده خود 5 احساس مختلف داریم.

1. آرام
2. خوشحال
3. غم انگیز
4. عصبانی
5. ترسناک

ما از کتابخانه Librosa در پایتون برای پردازش و استخراج ویژگی ها از فایل های صوتی استفاده کردیم. Librosa یک بسته پایتون برای تجزیه و تحلیل موسیقی و صدا است. این اجزای سازنده لازم برای ایجاد سیستم های بازیابی اطلاعات موسیقی را فراهم می کند. با استفاده از کتابخانه librosa ما توانستیم ویژگی های MFCC (ضریب فرکانس مل فرکانس Mel) را استخراج کنیم. MFCC ها ویژگی ای هستند که به طور گسترده در تشخیص خودکار گفتار و بلندگو استفاده می شود. ما همچنین صدای زنان و مردان را با استفاده از شناسه های ارائه شده در وب سایت جدا کردیم. این به این دلیل بود که در آزمایش ما دریافتیم که جداسازی صداهای زن و مرد 15 درصد افزایش یافته است. ممکن است به این دلیل باشد که صدای صدا بر نتایج تأثیر می گذارد. هر فایل صوتی ویژگی های زیادی را در اختیار ما قرار داد که اساساً

مجموعه ای از مقادیر مختلف بودند. این ویژگی ها توسط برچسب هایی که در مرحله قبل ایجاد کردیم ، اضافه شد.

گام بعدی شامل برخورد با ویژگی های مفقود شده برای برخی از فایل های صوتی است که طول آنها کوتاهتر است. ما میزان نمونه گیری را دو برابر افزایش دادیم تا ویژگی های منحصر به فرد هر گفتار احساسی را بدست آوریم. ما فرکانس نمونه برداری را حتی بیشتر افزایش ندادیم زیرا ممکن است سر و صدا را جمع آوری کرده و نتایج را تحت تأثیر قرار دهد.

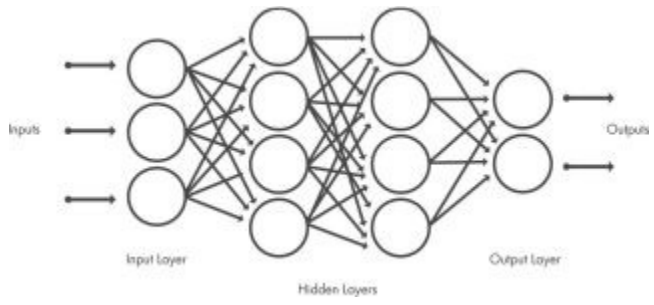
مراحل بعدی شامل تغییر داده ها ، تقسیم به قطار و آزمایش و سپس ساختن مدلی برای آموزش داده های ما است. ما یک مدل CNN ساختیم. MLP و LSTM مناسب نبودند زیرا دقت پایینی به ما می داد. از آنجا که پروژه ما یک مشکل طبقه بندی است که در آن احساسات مختلف دسته بندی شده است ، CNN برای ما بهترین کار را کرد.

Convolutional Neural Network:

یک شبکه عصبی پیچشی می تواند ده ها یا صدها لایه داشته باشد که هر کدام با تشخیص ویژگی های مختلف یک تصویر آشنا می شوند. فیلترها در هر تصویر آموزشی با وضوح مختلف اعمال می شوند و خروجی هر تصویر پیچیده به عنوان ورودی لایه بعدی استفاده می شود. فیلترها می توانند به عنوان ویژگی های بسیار ساده مانند روشنایی و لبه ها شروع شوند و پیچیدگی را به ویژگی هایی که به طور منحصر به فرد شی را تعریف می کنند ، افزایش دهند.

Feature Learning, Layers, and Classification

مانند سایر شبکه های عصبی ، CNN از یک لایه ورودی ، یک لایه خروجی و بسیاری از لایه های پنهان در بین آنها تشکیل شده است.



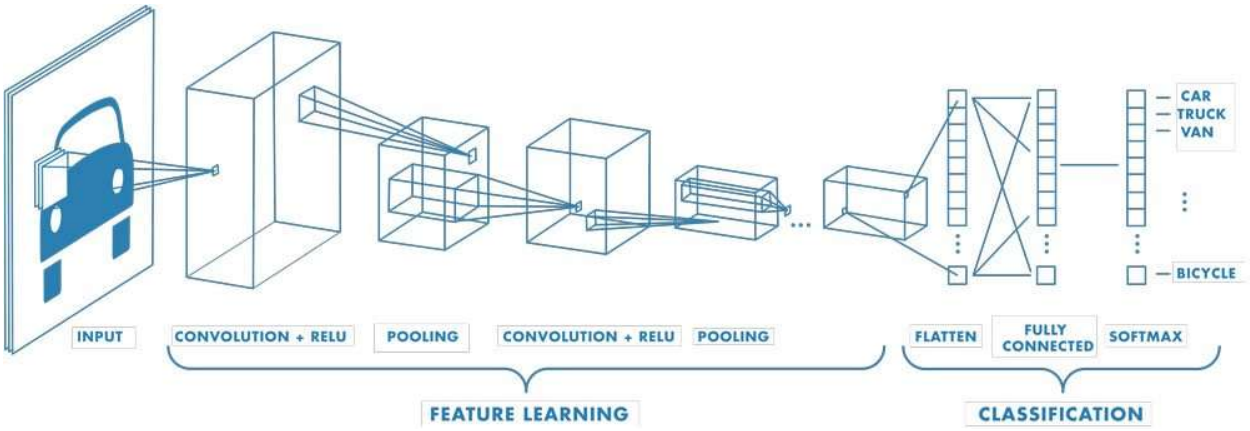
این لایه ها عملیات را انجام می دهند که داده ها را با هدف یادگیری ویژگی های خاص داده ها تغییر می دهد. سه مورد از رایج ترین لایه ها عبارتند از: کانولوشن ، فعال سازی یا ReLU و تجمع.

Convolution تصاویر ورودی را از طریق مجموعه ای از فیلترهای متحرک قرار می دهد که هر یک از آنها ویژگی های خاصی را از تصاویر فعال می کند.

واحد خطی تصحیح شده (ReLU) با ترسیم مقادیر منفی به صفر و حفظ مقادیر مثبت ، امکان آموزش سریعتر و م effectiveثرتر را فراهم می آورد. گاهی از این حالت به عنوان فعال سازی یاد می شود ، زیرا فقط ویژگی های فعال شده به لایه بعدی منتقل می شوند.

Pooling خروجی را با انجام نمونه گیری غیر خطی ساده کرده و تعداد پارامترهای مورد نیاز شبکه را برای یادگیری کاهش می دهد.

این عملیات در ده ها یا صدها لایه تکرار می شود و هر لایه یاد می گیرد که ویژگی های مختلف را تشخیص دهد.



تصویر بالا:

نمونه ای از شبکه با لایه های پیچشی متعدد. فیلترها در هر تصویر آموزشی با وضوح مختلف اعمال می شوند و خروجی هر تصویر پیچیده به عنوان ورودی لایه بعدی استفاده می شود.

وزن ها و سوگیری های مشترک

مانند شبکه عصبی سنتی ، CNN دارای نورون هایی با وزن و سوگیری است. مدل این ارزشها را در طول فرآیند آموزش می آموزد و با هر مثال آموزشی جدید ، آنها را به طور مداوم به روز می کند. با این حال ، در مورد CNN ها ، وزن و مقادیر تعصب برای همه نورونهای پنهان در یک لایه معین یکسان است.

این بدان معناست که همه نورون های پنهان یک ویژگی مشابه ، مانند لبه یا لکه را در مناطق مختلف تصویر تشخیص می دهند. این باعث می شود که شبکه نسبت به ترجمه اشیاء در یک تصویر تحمل کند. به عنوان مثال ، شبکه ای که برای تشخیص خودروها آموزش دیده است ، می تواند این کار را در هر کجا که ماشین در تصویر است انجام دهد.

لایه های طبقه بندی

پس از یادگیری ویژگی ها در لایه های مختلف ، معماری CNN به طبقه بندی تغییر می کند.

لایه بعدی تا آخر یک لایه کاملاً متصل است که بردار ابعاد K را خارج می کند که در آن K تعداد کلاس هایی است که شبکه قادر به پیش بینی آن است. این بردار شامل احتمالات برای هر کلاس از هر تصویر طبقه بندی شده است.

لایه نهایی معماری CNN از یک لایه طبقه بندی مانند softmax برای ارائه خروجی طبقه بندی استفاده می کند.

مدل CNN طبقه بندی بهتری بود. پس از آموزش مدلهای متعدد ، ما بهترین اعتبار validation accuracy ٪60 با 18 لایه ، عملکرد فعالسازی softmax ، عملکرد فعالسازی rmsprop ، اندازه دسته 32 و 1000 دوره را بدست آوردیم.

In [51]:
```
model.summary()
```

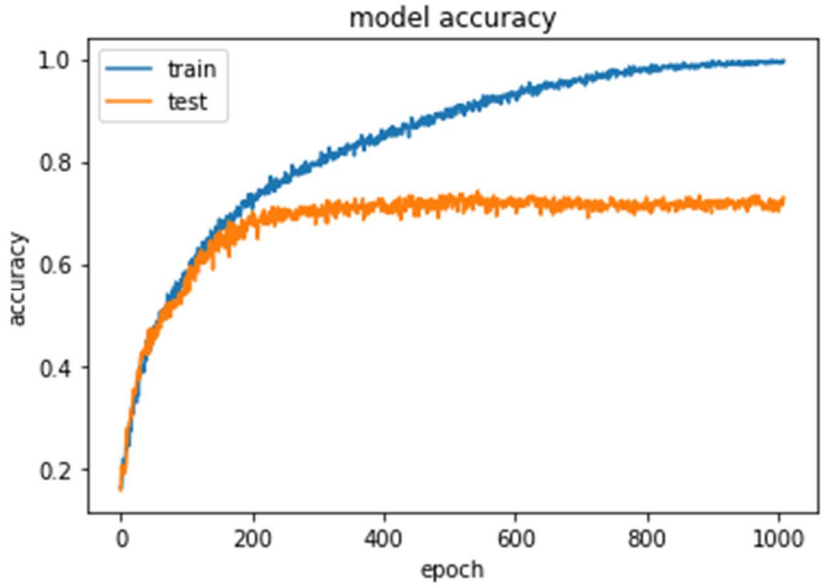
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d_1 (Conv1D)            (None, 216, 128)          768
_________________________________________________________________
activation_1 (Activation)    (None, 216, 128)          0
_________________________________________________________________
conv1d_2 (Conv1D)            (None, 216, 128)          82048
_________________________________________________________________
activation_2 (Activation)    (None, 216, 128)          0
_________________________________________________________________
dropout_1 (Dropout)          (None, 216, 128)          0
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 27, 128)           0
_________________________________________________________________
conv1d_3 (Conv1D)            (None, 27, 128)           82048
_________________________________________________________________
activation_3 (Activation)    (None, 27, 128)           0
_________________________________________________________________
conv1d_4 (Conv1D)            (None, 27, 128)           82048
_________________________________________________________________
activation_4 (Activation)    (None, 27, 128)           0
_________________________________________________________________
conv1d_5 (Conv1D)            (None, 27, 128)           82048
_________________________________________________________________
activation_5 (Activation)    (None, 27, 128)           0
_________________________________________________________________
dropout_2 (Dropout)          (None, 27, 128)           0
_________________________________________________________________
conv1d_6 (Conv1D)            (None, 27, 128)           82048
_________________________________________________________________
activation_6 (Activation)    (None, 27, 128)           0
_________________________________________________________________
flatten_1 (Flatten)          (None, 3456)              0
_________________________________________________________________
dense_1 (Dense)              (None, 10)                34570
_________________________________________________________________
activation_7 (Activation)    (None, 10)                0
=================================================================
Total params: 445,578
Trainable params: 445,578
Non-trainable params: 0
_________________________________________________________________
```

In [50]:

```
model = Sequential()

model.add(Conv1D(128, 5,padding='same',
                 input_shape=(216,1)))
model.add(Activation('relu'))
model.add(Conv1D(128, 5,padding='same'))
model.add(Activation('relu'))
model.add(Dropout(0.1))
model.add(MaxPooling1D(pool_size=(8)))
model.add(Conv1D(128, 5,padding='same',))
model.add(Activation('relu'))
model.add(Conv1D(128, 5,padding='same',))
model.add(Activation('relu'))
model.add(Conv1D(128, 5,padding='same',))
model.add(Activation('relu'))
model.add(Dropout(0.2))
model.add(Conv1D(128, 5,padding='same',))
model.add(Activation('relu'))
model.add(Flatten())
model.add(Dense(10))
model.add(Activation('softmax'))
opt = keras.optimizers.rmsprop(lr=0.00001, decay=1e-6)
```

In [110]:
```
#sigmoid
plt.plot(cnnhistory.history['acc'])
plt.plot(cnnhistory.history['val_acc'])
plt.title('model accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```



پس از آموزش مدل ، مجبور شدیم احساسات را در داده های آزمایش خود پیش بینی کنیم. تصویر زیر پیش بینی ما را با مقادیر واقعی نشان می دهد.

```
In [75]: finaldf[58:68]
```

Out[75]:

	actualvalues	predictedvalues
58	male_fearful	male_happy
59	male_fearful	male_fearful
60	male_fearful	male_fearful
61	male_fearful	male_fearful
62	male_sad	male_sad
63	male_fearful	male_fearful
64	male_happy	male_happy
65	female_angry	female_angry
66	female_angry	female_fearful
67	male_angry	male_angry

نتیجه گیری:

پس از ساخت مدل های مختلف متعدد ، ما بهترین مدل CNN خود را برای مشکل طبقه بندی احساسات خود پیدا کرده ایم. ما با مدل موجود خود به صحت اعتبار 70 درصد دست یافتیم. اگر داده های بیشتری برای کار داشته باشیم ، مدل ما می تواند عملکرد بهتری داشته باشد. آنچه بیشتر شگفت آور است این است که این مدل هنگام تشخیص صدای مرد و زن عالی عمل کرد. همچنین می توانیم در بالا نحوه پیش بینی مدل را در برابر مقادیر واقعی مشاهده کنیم. در آینده ما می توانیم دنباله ای به مدل توالی بسازیم تا صدا را بر اساس احساسات مختلف تولید کنیم. به عنوان مثال. صدای شاد ، صدای متعجب و غیره