

1. محاسبه احتمال :

قبل از بررسی سوال، در ابتدا باید یکسری از فرمول های مرتبط برای حل را بیان کرد:

- $P(c|d) = \frac{P(d|c)P(c)}{P(d)}$
 - $c_{MAP} = \underset{c \in C}{\operatorname{argmax}} P(d|c)P(c) = \underset{c \in C}{\operatorname{argmax}} P(x_1, x_2, \dots, x_n | c)P(c)$
 - $P(x_1, \dots, x_n | c) = P(x_1 | c) \cdot P(x_2 | c) \cdot P(x_3 | c) \cdot \dots \cdot P(x_n | c)$
- حال ما میدانیم که اگر به ازای ورودی X ، کلاس ما برابر با 1 شده باشد این به این منظور است که :

$$P(X|1) > P(X|0)$$

پس کافیت این مفاهیم را برای 4 عبارت اولیه که به ما داده شده است، توسعه دهیم :

$$P(aa|0) > P(aa|1) \rightarrow P(aa|0) = P(a|0) * P(a|0) > P(a|1) * P(a|1) \Rightarrow P(a|0) > P(a|1)$$

$$P(ab|0) > P(ab|1) \rightarrow P(ab|0) = P(a|0) * P(b|0) > P(a|1) * P(b|1)$$

$$P(ba|0) > P(ba|1) \rightarrow P(ba|0) = P(b|0) * P(a|0) > P(b|1) * P(a|1)$$

$$P(bb|0) < P(bb|1) \rightarrow P(bb|0) = P(b|0) * P(b|0) > P(b|1) * P(b|1) \Rightarrow P(b|0) > P(b|1)$$

- $P(1) = 0.75$
- $P(0) = 0.25$
- $P(a|0) = 5/8$
- $P(b|0) = 3/8$
- $P(a|1) = 1/4$
- $P(b|1) = 3/4$
- aabaa 0: $(3/4) * (5/8) * (5/8) * (3/8) * (5/8) * (5/8) = 0.042$
 1: $(1/4) * (1/4) * (1/4) * (3/4) * (1/4) * (1/4) = 0.0007$

\leq عبارت بالا مربوط به کلاس 0 است

- b 0: $(3/4) * (3/8) = 0.28$
 1: $(1/4) * (3/4) = 0.18$

\leq عبارت بالا مربوط به کلاس 0 است

- bba 0: $(3/4) * (3/8) * (3/8) * (5/8) = 0.065$
 1: $(1/4) * (3/4) * (3/4) * (1/4) = 0.035$

\leq عبارت بالا مربوط به کلاس 0 است

- bbbb 0: $(3/4)*(3/8)*(3/8)*(3/8)*(3/8) = 0.015$
 1: $(1/4)*(3/4)*(3/4)*(3/4)*(3/4) = 0.08$

=> عبارت بالا مربوط به کلاس 1 است

2. توضیح در مورد هر دیتاست:

MNIST: پایگاه داده MNIST (پایگاه داده مؤسسه ملی استاندارد و فناوری اصلاح شده) یک پایگاه داده بزرگ از ارقام دست‌نویس است که معمولاً برای آموزش سیستم‌های مختلف پردازش تصویر استفاده می‌شود. پایگاه داده همچنین به طور گسترده برای آموزش و آزمایش در زمینه یادگیری ماشین استفاده می‌شود. با "آمیختن مجدد" نمونه‌های مجموعه داده‌های اصلی NIST ایجاد شد. سازندگان احساس کردند که از آنجایی که مجموعه داده آموزشی NIST از کارمندان اداره سرشماری آمریکا گرفته شده است، درحالی که مجموعه داده آزمایشی از دانش‌آموزان دبیرستانی آمریکایی گرفته شده است، برای آزمایش‌های یادگیری ماشین مناسب نیست.

علاوه بر این، تصاویر سیاه‌وسفید از NIST نرمال‌سازی شدند تا در یک جعبه محدودکننده پیکسل ۲۸ در ۲۸ قرار بگیرند و ضد مستعار شوند که سطوح خاکستری را معرفی می‌کرد.

پایگاه داده MNIST شامل 60000 تصویر آموزشی و ۱۰۰۰۰ تصویر آزمایشی است. نیمی از مجموعه آموزشی و نیمی از مجموعه تست از مجموعه داده آموزشی NIST گرفته شده است، درحالی که نیمی دیگر از مجموعه آموزشی و نیمی دیگر از مجموعه تست از مجموعه داده‌های آزمایشی NIST گرفته شده است.

CIFAR-10: مجموعه داده CIFAR-10 (مؤسسه تحقیقات پیشرفته کانادایی) مجموعه‌ای از تصاویر است که معمولاً برای آموزش الگوریتم‌های یادگیری ماشین و بینایی کامپیوتر استفاده می‌شود. این یکی از پرکاربردترین مجموعه داده‌ها برای تحقیقات یادگیری ماشینی است. مجموعه داده CIFAR-10 شامل ۶۰۰۰۰ تصویر رنگی ۳۲ در ۳۲ در ۱۰ کلاس مختلف است. ۱۰ کلاس مختلف نشان‌دهنده هواپیما، اتومبیل، پرنده، گربه، گوزن، سگ، قورباغه، اسب، کشتی و کامیون است. ۶۰۰۰ تصویر از هر کلاس وجود دارد.

FER-2013: حاوی تقریباً 30000 تصویر RGB صورت با عبارات مختلف با اندازه محدود به 48x48 است و برچسب‌های اصلی آن را می‌توان به ۷ نوع تقسیم کرد:

عصبانی، انزجار، ترس، شاد، غمگین، سورپرایز، خنثی.

برچسب انزجار حداقل ۶۰۰ تعداد تصویر را دارد، درحالی که برچسب‌های دیگر هرکدام نزدیک به ۵۰۰۰ نمونه دارند.

3. to categorical

در MNIST و CIFAR-10 ما از to categorical برای تبدیل داده‌های آموزشی خود قبل از اینکه به مدل خود منتقل کنید، استفاده می‌کنید. اگر داده‌های آموزشی ما از رده‌ها به عنوان اعداد استفاده می‌کند، to categorical آن اعداد را در بردارهای مناسب برای استفاده با مدل‌ها تبدیل می‌کند و ما نمی‌توانید به سادگی یک مدل طبقه‌بندی را بدون آن آموزش دهید.

4. x_train و y_train:

در هر دیتاست ما نیازمند یک تصویر و برچسب آن هستیم. اصولاً در x_train مقدار عکس‌ها ذخیره می‌شوند و در y_train مقدار برچسب‌ها قرار دارند.

5. ImageDataGenerator:

این به ما امکان می‌دهد تا زمانی که مدل شما هنوز در حال آموزش است، تصاویر خود را در real-time افزایش دهید! ما می‌توانید هر گونه تبدیل تصادفی را روی هر تصویر آموزشی که به مدل منتقل می‌شود اعمال کنید. این نه تنها مدل شما را قوی می‌کند، بلکه در حافظه سربرابر نیز صرفه جویی می‌کند! و همین‌طور به ما امکان تولید دسته‌ها را می‌دهد: می‌توانید از تکه‌های کوچک‌تری از داده‌ها برای آموزش مدل خود با استفاده از نزول گرادیان دسته‌ای استفاده کنید. و عملاً همه مجموعه داده‌های دنیای واقعی مربوط به یادگیری عمیق معمولاً نمی‌توانند یکباره در حافظه جای بگیرند.

6. Summary ما نوع لایه‌ها، تایپ آن‌ها را و همین‌طور نام‌های مشخص می‌کند و علاوه بر آن موارد در وسط

آن، مشخص می‌کند که چه shape را به عنوان خروجی تولید می‌کند. همین‌طور Param را نیز مشخص می‌کند که به طور مثال در حالتی که لایه ما HiddenLayer است از لایه قبلی 2500 ورودی به همراه bias دریافت می‌کند (2500+1) که چون در این لایه 128 تا unit داریم مقدار پارام‌ها برای این لایه برابر با $128 * 2501$ میشود.

همین‌طور چون مقدار pre-defined شده‌ای نداریم، output‌های ما none هستند.

- Dense Layers:
 - $output_size * (input_size + 1) == number_parameters$
- For Conv Layers:
 - $output_channels * (input_channels * window_size + 1) == number_parameters$

Layer (type)	Output Shape	Param #
InputLayer (Flatten)	(None, 2500)	0
HiddenLayer (Dense)	(None, 128)	320128
relu (Activation)	(None, 128)	0
OutputLayer (Dense)	(None, 5)	645
softmax (Activation)	(None, 5)	0
Total params: 320,773		
Trainable params: 320,773		
Non-trainable params: 0		

7. در حالت کلی نباید انتظار داشت مدل به صورت خوبی عمل کند! برای این موضوع میتوان به دلایل زیر اشاره کرد :

- مدل ما تنها دارای یک لایه مخفی است و در این حالت که از دیتاست FER-2013 استفاده میکنیم، برعکس حالت MNIST، تصاویر ما دارای ویژگی های بیشتری هستند که آنها را از یکدیگر متمایز میکنند و به همین دلیل نیاز داریم از تعداد لایه ها و تعداد نوروں های بیشتری برای بهبود عملکرد بهره ببریم.
- تعداد epoch های آموزش بسیار پایین است.
- با این حال در حالت کلی برای آموزش یه NN برای این نوع تصاویر پیچیده که مدل ما باید چندین ویژگی را متمایز کنه، باید از CNN ها بهره ببریم و حتی با افزایش تعداد لایه های میانی و epoch به احتمال زیاد، عملکرد قابل قبولی نخواهند داشت.