

Sign Language Recognition

پروژه هوش محاسباتی استاد علی تورانی

دستیار استاد: رضا خان محمدی

گروه 2: امیرحسین غمگسار – 9712268105

علی خورسند جلالی – 960122680104

دیتاست (dataset)

دیتاست اولیه شامل عکس‌هایی از 5 شخص بود و هر کدام از این کلاس‌ها دارای کلاس‌هایی از حروف الفبای انگلیسی به جز حروف “j” و “z” بودند، بر خلاف سایر حروف امکان نمایش این دو حرف توسط تنها یک فریم وجود ندارد. این دیتاست شامل بیش از 65,000 عکس در مجموع بود که از دارای زوایا و پس زمینه‌های مختلفی بودند. ([لینک دانلود](#))

دیتاست مورد استفاده ما دارای 1,000 عکس برای هر حرف می‌باشد که این عکس‌ها به صورت تصادفی از میان تمامی عکس‌ها و اشخاص انتخاب شدند. از میان 1,000 عکس، 400 عکس برای train، 400 عکس برای validation و 200 عکس برای test برگزیده شدند. لینک:

[پارت 1](#) - [پارت 2](#) - [پارت 3](#) - [پارت 4](#) - [پارت 5](#) - [پارت 6](#) - [پارت 7](#) - [پارت 8](#)



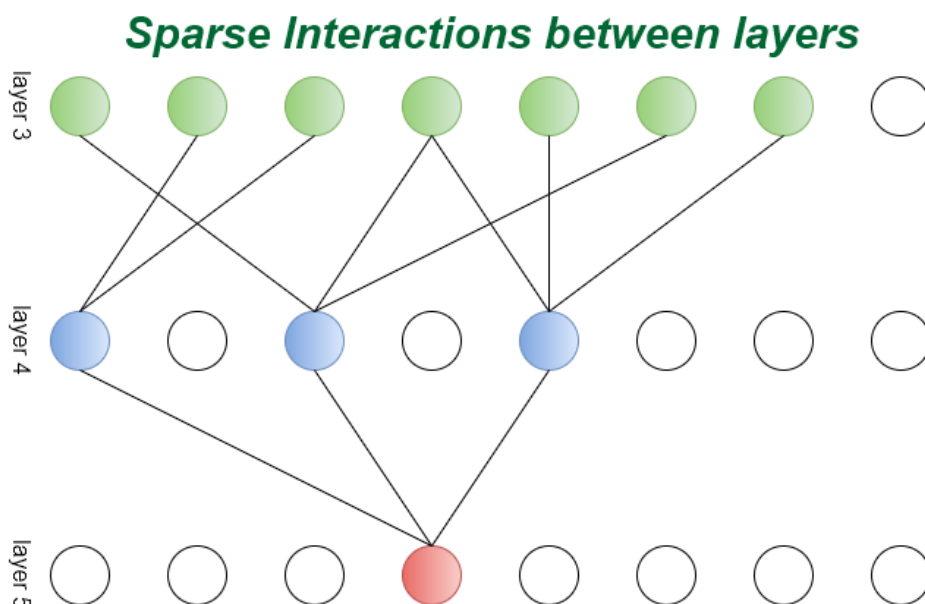
Conv2D

شبکه‌های عصبی کانولوشن دارای 3 خصیصه بنیادی هستند که باعث کاهش تعداد پارامترها در شبکه عصبی می‌شوند:

- تعامل‌های پراکنده بین لایه‌ها (Sparse Interactions):

در شبکه‌های عصبی feed-forward معمول، هر یک از نورون‌های یک لایه به تمام نورون‌های موجود در لایه بعدی متصل است و با آن ارتباط مستقیم دارد که این کار باعث ایجاد پارامترهای بسیاری می‌شود که باید توسط شبکه عصبی یاد گرفته شود که این نیز ممکن است باعث ایجاد مشکلات دیگری مانند نیاز به training data زیاد، افزایش convergence time و به دست آوردن یک مدل overfitted شود.

شبکه‌های عصبی کانولوشن (CNN) می‌تواند باعث کاهش تعداد پارامترها از طریق تعامل غیرمستقیم شود و هر نورون فقط به نورون‌هایی در لایه بعد متصل می‌شود که تحت تاثیر آن به صورت مستقیم باشند.



- اشتراک گذاری پارامترها (Parameter Sharing):

نقاط منحصر به فردی که در Feature map دارای عمق (depth) یکسانی هستند که 3D volume نام دارد توسط کرنل یکسانی ساخته شده‌اند یا به عبارتی دیگر با استفاده از مجموعه‌ای از پارامترهای به اشتراک گذاشته شده یکسانی ساخته شده‌اند که باعث کاهش تعداد پارامترها می‌شود.

- Equivariant Representation:

به عنوان مثال اگر دو عمل convolution و equivariant translation باشند، دو ترتیب مختلف اجرای این دو عمل به یک نتیجه می‌انجامد. این کار برای اشتراک گذاری پارامترها میان داده‌ها استفاده می‌شود، برای مثال در در داده‌های از نوع عکس، اولین لایه کانولوشن معمولاً بر روی تشخیص لبه (edge detection) تمرکز می‌کند و ممکن است لبه‌های مشابه‌ای در عکس مشاهده شود، پس استفاده از پارامترهای یکسان برای این لبه‌ها منطقی و مفید است.

لایه‌های CNN:

- لایه کانولوشن:

معمولاً اولین لایه CNN است که دیتا را با استفاده از فیلترها یا کرنل‌ها Convolve می‌کند. سایز فیلتر همان عمق ورودی برنامه است. عمل کانولوشن به عنوان مثال با 9 بار تکرار ضرب فیلتر:

1	0	1
0	1	0
1	0	1

3*3 filter

1 _{*1}	1 _{*0}	1 _{*1}	0	0
0 _{*0}	1 _{*1}	1 _{*0}	1	0
0 _{*1}	0 _{*0}	1 _{*1}	1	1
0	0	1	1	0
0	1	1	0	0

image

4	3	4
2	4	3
2	3	4

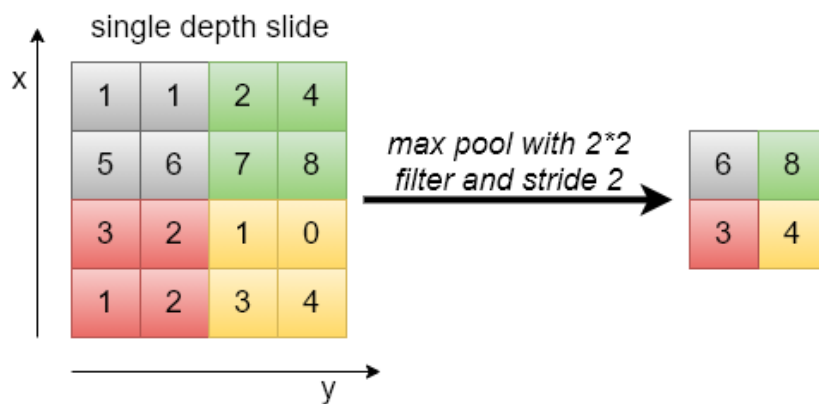
Convolved feature

- تابع فعال ساز (activation)

فقط لایه‌های غیر خطی در میان لایه‌های متوالی کانوولوشن مورد استفاده قرار می‌گیرند، در صورتی که فقط از تابع های فعال ساز خطی استفاده شود، یادگیری وجود ندارد.

- Pooling:

این کار شامل کاهش ابعاد خصیصه‌ها می‌شود تا پارامترهای کمتری در طول عمل training صورت گیرد، دو ابر پارامتر در این لایه معرفی می‌شوند، ابعاد وسعت فضایی که مقداری است که می‌توان n خصیصه را به یک مقدار map کرد و پارامتر Stride، اندازه گام‌هایی را که می‌توان در عرض یا طول تصویر برداشت مشخص می‌کند و باعث کاهش احتمال overfitting می‌شود. یک فیلتر max pooling بیشترین مقدار در میان ناحیه مورد نظر را بر می‌گرداند، در حالی که فیلتر average pooling میانگین آنها را برمی‌گرداند، ولی عمل max pooling کارایی بهتری دارد. عمل Pooling باعث تغییر عمق نمی‌شود.

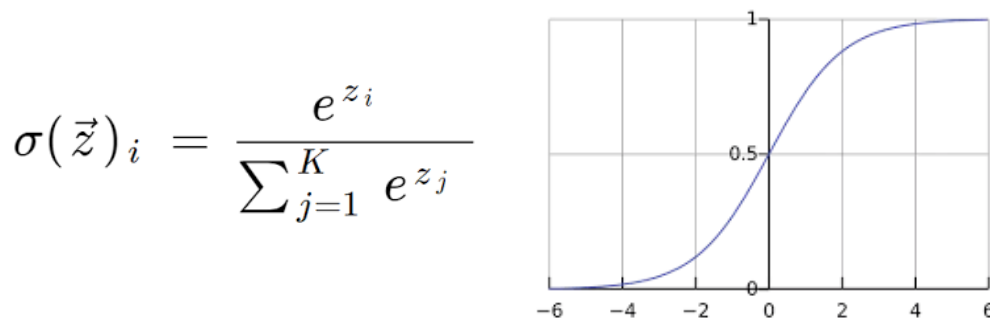


• Fully Connected Layer (Dense Layer)

معمولا راهی سبک برای یادگیری ترکیبات غیر خطی خصیصه‌های خروجی از لایه‌های کانوولوشن است. لایه‌های کانوولوشن خصیصه‌های فضایی ثابت، معنی‌دار و دارای ابعاد کم را تهیه می‌کنند و لایه‌های Fully Connected، تابعی احتمالا غیر خطی را در آن فضا یاد می‌گیرند. خروجی maxPooling که در برنامه ما 2D است با استفاده از عمل Flattening به ورودی مورد نظر و 1D لایه‌های FC (Fully Connected) تبدیل می‌شود. در ادامه برای طبقه‌بندی نیاز به ایجاد لایه‌های پنهانی و اعمال کردن تابع فعال‌ساز softmax برای آخرین لایه از نورون‌ها است.

تابع softmax

این تابع در machine learning برای تبدیل خروجی یک لایه از شبکه عصبی به احتمالات است که این احتمالات می‌توانند برای دسته‌بندی نوعی خاص از عکس باشند.



تابع فعال‌ساز relu

این تابع که نام آن به معنای rectified linear unit است، یکی از محبوب‌ترین تابع‌های فعال‌ساز در CNNهاست که وزن‌های کوچکتر از 0 را به 0 تبدیل می‌کند.

$$f(w_i) = \max(0, w_i)$$

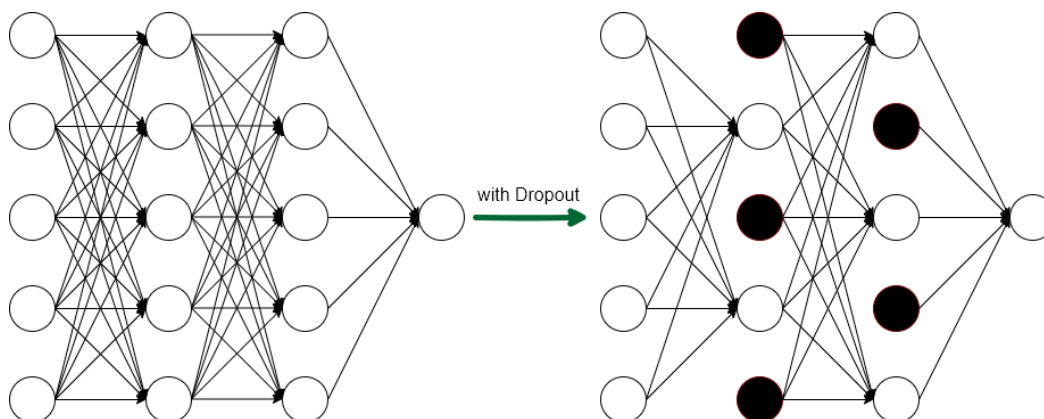
Batch Normalization

به فرایند نرمال سازی تابع فعال ساز برای تمامی لایه ها می گویند و تفاوت آن با نرمال سازی معمولی این است که در این نوع نرمال سازی در هر مرحله به جای یه دیتا (عکس)، دسته ای از دیتا به شبکه داده می شود تا وزن ها آپدیت شوند. فواید آن به شرح زیر است:

- افزایش سرعت training به دلیل loss smoothing که این هموارسازی باعث رفتار قابل پیش بینی تر و پایدارتری از سطوح شیب دار می شود.
- کاهش اهمیت وزن اولیه و شروع بهینه تر
- مرتب کردن مدل تا حدود کمی

Dropout

برای جلوگیری از overfitting مورد استفاده قرار می گیرد، کاری مشابه regularization انجام می دهد، به این صورت که برای هر داده در شبکه عصبی به عنوان مثال نصف (در برنامه این احتمال 0.2 است) تابع های فعال ساز (Activations) را به طور کاملاً تصادفی صفر می کند و در نتیجه نصف داده های جاری در شبکه از بین می روند. در این حالت شبکه نمی تواند بر یک activation خاص تکیه کند، زیرا ممکن است توسط Dropout از بین برود، بنابراین activation های دیگری وجود دارند تا وظایف مورد نظر را به انجام برسانند. پس Dropout باعث جلوگیری از به روز رسانی همه ی نورون های یک لایه به طور همزمان می شود تا همه ی این تغییرات وزن برای رسیدن به یک هدف منحصر به فرد نباشند و همچنین باعث می شود تا activation های لایه های پنهانی، پراکنده (Sparse) شوند که یک خصوصیت مطلوب می باشد.



لایه‌های شبکه عصبی مورد استفاده:

