Redes neuronales convolucionales para clasificar imágenes

Para clasificar imágenes vamos a utilizar una red neuronal un poco especial: la red neuronal convolucional, también llamada CNN (*Convulational Neural Network* o *ConvNet*).

Este algoritmo de aprendizaje es similar a un pastel milhojas porque está compuesto de muchas capas. Las últimas capas (parte superior del milhojas) están formadas por redes neuronales como las que hemos descubierto en los capítulos anteriores. Es decir, sus datos de entrada proceden de las capas anteriores, llamadas capas de convoluciones y cuyo principio vamos a descubrir a continuación.

1. Necesitamos muchos datos de aprendizaje

Para que la red de neuronas pueda realizar una buena clasificación de imágenes, hay que darle una base de aprendizaje. Pero cuando se trata de imágenes, su cantidad puede convertirse en ingente con mucha rapidez.

En efecto, nuestro cerebro humano puede realizar la clasificación entre un gato y un perro, ya sea negro, blanco, esté de cara, de perfil, de espalda, en pleno día, en penumbra y en distintos entornos y posiciones posibles. Así que podemos hacernos una idea de la cantidad de imágenes necesarias.

Si desea iniciarse en un proyecto personal de clasificación de imágenes, eso también le permitirá apreciar el trabajo de fotografía que se debe realizar. Por suerte para nosotros, hay conjuntos de observaciones disponibles en Internet para entrenarnos.

Para poder realizar su aprendizaje, la máquina tendrá que manipular la imagen. Entramos entonces en el campo del tratamiento de imágenes que, como seguramente ya sabe, consume recursos materiales. Por lo tanto, realizar la clasificación de imágenes necesita una cantidad de tiempo importante para la creación del conjunto de aprendizaje, así como inversión en el material para llevar a cabo el aprendizaje.

2. Una herramienta para ilustrar nuestras palabras

Las redes neuronales convolucionales están compuestas de varias capas de tratamientos que se pueden esquematizar con facilidad. Gracias a Adam W. Harley, podemos ir más lejos en la visualización del interior de las redes neuronales convolucionales, ayudados por una herramienta interactiva disponible en esta dirección: <http://scs.ryerson.ca/~aharley/vis/conv/flat.html>

Interfaz de usuario gráfica

Descripción generada automáticamente

*Una herramienta de visualización para comprender las redes neuronales convolucionales*

El principio de esta herramienta es bastante sencillo. En la parte superior izquierda vamos a dibujar un número de 0 a 9 y la máquina será la encargada de clasificar este dibujo como número. En la parte derecha, podremos visualizar todas las etapas sucesivas que realiza la máquina para lograr la clasificación. Cada etapa debe ser considerada como una capa del algoritmo de clasificación.

Imagen de la pantalla de un video juego

Descripción generada automáticamente con confianza media

*Descripción de la interfaz de la herramienta que permite comprender el funcionamiento de las redes neuronales convolucionales*

Con algunas resoluciones de pantalla la parte izquierda no se ve al completo. En ese caso, solo tiene que alejar un poco la escena utilizando la opción de zoom de los navegadores web.

3. La imagen de entrada

Las redes neuronales convolucionales sirven para clasificar imágenes, así que todo parte de una imagen de origen. Para crearla dibujamos un numero entre 0 y 9 en la parte superior de la izquierda de la pantalla y observamos lo que pasa en la zona de la derecha.

Lo primero que observamos es que nuestra imagen de entrada (input layer) se encuentra en la parte inferior de la pantalla y se ve dividida en cuadrados pequeños. Por supuesto, estos cuadrados son los píxeles de la imagen.

Si vamos subiendo poco a poco, también vemos que han aparecido otras imágenes hasta obtener la predicción del número dibujado (en nuestro caso es el 1).

Una pantalla de un video juego

Descripción generada automáticamente con confianza media

*Clasificación del número 1*

4. Características

Ahora vamos a ver el concepto de característica. Imaginemos que quisiéramos hacer una clasificación de una imagen que contiene un conejo. Nuestro objetivo es que, cuando le presentemos una imagen de un conejo a nuestra máquina, indique si se trata de un conejo.

Una de las soluciones más intuitivas que se nos ocurren para resolver este problema es comparar dos imágenes píxel por píxel, una imagen de aprendizaje y una imagen pendiente de clasificar. Si las dos imágenes se corresponden, entonces se puede afirmar que se trata de un conejo.

Sin embargo, esto tiene un inconveniente, y es que la imagen que se ha de clasificar debe ser completamente idéntica a la imagen de referencia, lo cual resulta bastante imposible en los casos reales de aplicación.

La solución para resolver este problema consistirá en capturar características en la imagen origen (fragmentos de imágenes) y buscarlas en la imagen pendiente de clasificar. Es lo que se llama «principio de convolución», que vamos a descubrir ahora.

Foto montaje de un gato y un perro

Descripción generada automáticamente con confianza media

*Esta ilustración sirve de ejemplo para mostrar la necesidad de utilizar características para la clasificación de imágenes.*

5. La convolución

Como acabamos de ver, vamos a utilizar características de una imagen para intentar encontrarlas en otra imagen pendiente de clasificar. Si la imagen pendiente de clasificar contiene una gran cantidad de características comunes con la imagen de origen de clasificación, hay muchas probabilidades de que las imágenes sean muy similares. La fase de aprendizaje consistirá en extraer estas características para, a continuación, alimentar una red neuronal de modo que aprenda a establecer una relación entre las características de la imagen de origen acopladas a un etiquetado (nombre del objeto que se ha de reconocer). Al final del aprendizaje, la red de neuronas podrá decir «He aprendido que si la presencia de estas características es mayoritaria, entonces la imagen que me presentan es un conejo».

Para aclarar este concepto, consideremos una imagen con dimensiones de 5 píxeles por 5 píxeles y una característica de 3 píxeles por 3 píxeles. Esta característica se denomina filtro o kernel. Los píxeles negros de la imagen tendrán el valor 1; los blancos, 0, y los grises, -1.

Tabla

Descripción generada automáticamente

*Extracción de una característica de una imagen*

La convolución consistirá en aplicar el filtro sobre el conjunto de la imagen para deducir una imagen nueva.

Las cuatro imágenes siguientes explican el procedimiento:

* Se coloca el filtro en la parte superior izquierda de la imagen.
* Cada valor de píxel de la imagen que se sitúa en el filtro se multiplica por el valor del píxel del filtro.
* Se efectúa la suma de estos productos.
* Se desplaza un píxel y se vuelve a empezar el producto y la suma. Este desplazamiento se llama *Stride* y, en nuestro caso, tiene valor 1.

Se realiza este proceso hasta que se haya filtrado el conjunto de la imagen. Así hemos creado una imagen nueva más pequeña que la imagen de origen y que tiene valores nuevos, como muestra la tabla que aparece a continuación. La finalidad de estos valores es destacar algunas características de la imagen.

Forma, Calendario

Descripción generada automáticamente

*Característica procedente de la convolución*

Durante la puesta en práctica, veremos que en la fase de convolución vamos a especificar un método de activación. En la mayoría de los casos, elegimos el método ReLU, que permite eliminar los valores negativos de la imagen de convolución sustituyéndolos por el valor 0. Sin esta activación, los resultados de aprendizaje no serían los esperados.

Calculadora con pantalla y botones

Descripción generada automáticamente con confianza media

*Etapas de convoluciones*

Calculadora con pantalla y botones

Descripción generada automáticamente con confianza baja

*Etapas de convoluciones*

Calendario

Descripción generada automáticamente

*Etapas de convoluciones*

Calculadora color gris

Descripción generada automáticamente con confianza media

*Etapas de convoluciones*

Podemos observar que en la convolución n.º 8 la suma es mayor que la de las otras convoluciones. Esto se debe al hecho de que el filtro aplicado corresponde perfectamente a la parte de la imagen que está siendo analizada; por lo tanto, hay un punto de concordancia.

Ahora volvemos a nuestra herramienta de reconocimiento de números. Se puede observar que en la parte superior de la imagen de entrada hay otras 6 imágenes. Estas imágenes proceden de la convolución de 6 filtros.

A la derecha está la imagen procedente del filtro n.º 1, y a la izquierda, la procedente del filtro n.º 6. El tamaño de los filtros aplicados es de 5 píxeles \* 5 píxeles y por supuesto son diferentes.

Si pasa el ratón sobre un píxel de la imagen originada a partir de la convolución, podrá ver el fragmento de la imagen de origen a partir del que ha sido creado.

Un luz de freno

Descripción generada automáticamente

*Principio de convoluciones*

Como acabamos de ver, la cantidad de filtros utilizados corresponde a las características que hay que aprender. Entonces tendríamos tendencia a decir que, cuantos más filtros distintos se utilizan en la fase de aprendizaje, más eficaz será dicha fase. Sin embargo, eso puede causar un sobreajuste del algoritmo y tener un efecto inverso al esperado.

Desgraciadamente, no existe ningún método milagroso para determinar la cantidad de filtros que hay que utilizar ni su dimensión. A menudo es necesario definir estos valores de manera empírica.

Si aplicamos este tratamiento de convolución a nuestra imagen del conejo (ver a continuación), podemos darnos cuenta de que la cantidad de cálculos puede aumentar mucho con rapidez si el tamaño de la imagen es grande y si se multiplica la cantidad de filtros. En ese caso este cálculo demanda muchos recursos de procesador y de tarjeta gráfica. Veremos que en nuestro caso práctico vamos a configurar capas de convoluciones de 128 filtros con un tamaño de 3 píxeles por 3 píxeles. Por eso, en la mayoría de los casos se utilizan imágenes pequeñas (28 x 28 píxeles o 32 x 32 píxeles). Hay que destacar que algunos fabricantes de tarjetas gráficas (NviDia) se han especializado en la realización de material dedicado a Deep Learning, con productos que sobrepasan los 2000 € para las tarjetas de mayor rendimiento.

Breve nota técnica: En términos de tratamiento de imagen, es preferible trabajar con imágenes en escala de grises. En efecto, una imagen en color contiene 3 capas de colores (rojo, verde y azul), mientras que una imagen de escala de grises solo contiene una. Esta característica mejora el tiempo de tratamiento porque en lugar de tratar 3 capas solo debemos tratar una.

Imagen en blanco y negro de un gato

Descripción generada automáticamente con confianza media

*Convolución aplicada a una imagen grande*

6. Pooling

Ahora que disponemos de una imagen filtrada, vamos a aplicarle un tratamiento nuevo denominado Pooling, que permite extraer las características importantes procedentes de la convolución.

Este tratamiento consiste en desplazar una ventana dentro de la imagen procedente de la convolución. En el marco de un MaxPooling se busca el valor máximo contenido en el interior de esta ventana; en el marco de un Average Pooling se calcula la media de los valores contenidos dentro de la ventana, y finalmente en el marco de un Pooling Stochastique se retiene un valor en función de estimaciones probabilísticas.

Retomemos la imagen procedente de nuestra convolución y apliquémosle una ventana de *pooling* de 2 píxeles por 2 píxeles con un stride (desplazamiento) de 1, al igual que hemos hecho en la etapa de convolución:

Tabla

Descripción generada automáticamente

Forma, Tabla

Descripción generada automáticamente

Tabla

Descripción generada automáticamente con confianza baja

Forma, Tabla

Descripción generada automáticamente con confianza media

Lo que nos da una nueva imagen que solo contiene los siguientes valores destacados:

Tabla, Calendario

Descripción generada automáticamente

Como podemos comprobar, después de pasar por una convolución y un *pooling*, nuestra imagen de origen, que tenía 5 x 5 píxeles, se ha reducido a una imagen de 2 x 2 que conserva las características de la imagen procedente de la convolución.

Esta reducción de tamaño es especialmente beneficiosa para los distintos cálculos que haremos después.

Si observamos la capa de *pooling* en la herramienta de clasificación de los números dibujados manualmente, comprobamos que cada capa de convolución da lugar a un *pooling*. El *pooling* utilizado para cada filtro es un MaxPooling con un tamaño de 2 x 2 y un desplazamiento de dos (*stride* en inglés).

Pantalla de un video juego

Descripción generada automáticamente con confianza media

*Ilustración del principio de pooling*

7. Múltiples capas de convoluciones

Como podemos comprobar en la herramienta de predicción de números dibujados manualmente, la capa de *pooling* que hemos visto antes está conectada a una capa de convolución, que a su vez también está conectada a una capa de *pooling*.

Esto nos da las siguientes capas de convolución:

* Una primera capa de convolución compuesta de 6 filtros de tamaño 5 x 5 con un *stride* igual a 1.
* Una capa de *pooling* de tamaño 2 x 2.
* Una segunda capa de convolución compuesta de 16 filtros de tamaño 5 x 5 con un *stride* igual a 1.
* Una capa de *pooling* de tamaño 2 x 2.

El objetivo es reducir la imagen conservando la información importante.



*Les distintas capas de convoluciones y de pooling.*

Los parámetros de las distintas capas utilizadas en la herramienta de visualización están disponibles en el documento PDF «An Interactive Node-Link Visualization of Convolutional Neural Networks», escrito por Adam W. Harley y que se puede descargar en esta dirección: <http://scs.ryerson.ca/~aharley/vis/harley_vis_isvc15.pdf>

8. Aplanar (Flatten)

Tras concluir las distintas operaciones de convoluciones y de *pooling*, vamos a preparar los datos para que se puedan integrar en una red neuronal.

Esta etapa de preparación consiste en «aplanar» las distintas imágenes resultantes de las operaciones de *pooling* dándoles forma de una sola columna (o línea dentro de la herramienta). Entonces creamos un vector:

Pantalla de computadora

Descripción generada automáticamente

*«Aplanado» de las distintas imágenes (Flatten)*

Imagen que contiene Diagrama

Descripción generada automáticamente

*Fase de aplanado (Flatten)*

9. Aprendizaje

Finalmente, la última fase de la red neuronal convolucional es el aprendizaje propiamente dicho con la ayuda de capas de neuronas completamente conectadas. La entrada de estas capas es la capa de aplanado anteriormente realizada, y la salida es una predicción.

El funcionamiento del aprendizaje es idéntico al que hemos descubierto en el capítulo anterior: definición de los pesos, predicción, retropropagación…

Al mirar las últimas capas de la herramienta de predicción, podemos ver que la red neuronal permite predecir una probabilidad por categoría de número (de 0 a 9).

Según el autor de la herramienta, la red neuronal está compuesta por 784 neuronas en la entrada, 300 neuronas en la primera capa oculta, 100 en la segunda y 10 en la salida, correspondientes a cada uno de los distintos números.

Interfaz de usuario gráfica

Descripción generada automáticamente

*Predicción emitida por la red neuronal*

10. Esquema global

Para resumir todo lo que acabamos de ver, podemos utilizar un esquema muy popular representado por la siguiente figura (<https://www.mathworks.com/videos/introduction-to-deep-learning-what-are-convolutional-neural-networks--1489512765771.html>), donde aparecen todas fases de una red neuronal convolucional.

Diagrama, Dibujo de ingeniería

Descripción generada automáticamente

*Funcionamiento de una red convolucional (fuente mathworks)*