Paper: Backpropagation Autor: Santiago Ramírez Moreno

Se proponen varias mejoras en el procedimiento de propagación hacia atrás para aumentar la velocidad de entrenamiento, y discutimos sus limitaciones con respecto al rendimiento de generalización. La superficie del terror está modelada para evitar mínimos locales y áreas planas. Los pesos sinápticos se actualizan con la mayor frecuencia posible. Tanto el tamaño del paso como el impulso se escalan dinámicamente a los valores más grandes posibles que no se resaltan en el rebasamiento

Recientemente, el uso del nuevo procedimiento de aprendizaje y la disponibilidad de supercomputaciones rápidas han permitido utilizar arquitecturas conexionistas para reconocer patrones del "mundo real". Por ejemplo, se ha obtenido un rendimiento muy alto con las redes neuronales de retardo de tiempo, que se entrenan utilizando la propagación inversa. Sin embargo, estos altos resultados se lograron a expensas de la velocidad de entrenamiento, lo que podría ser un obstáculo para aprender una gran base de datos. Surge un problema de escala y es útil para evaluar el tiempo de aprendizaje en función de las dimensiones de la tarea y la red. Si denotamos el número de conexiones w, el número de muestras de entrenamiento my el número promedio de veces que uno tiene que presentar cada muestra de entrenamiento para que aprenda correctamente p, entonces el tiempo de aprendizaje t = p.w.m (en número de conexiones que se procesan).

Presentamos aquí la red con la arquitectura a.r.d la base de datos. Un TDNN es un perpecptron multicapa (MLP) que incluye una dimensión de tiempo: dos unidades físicas están conectadas a través de varios pesos, cada uno correspondiente a un retraso diferente en el tiempo. Es posible transformar esta dimensión de tiempo en una dimensión espacial. Cada unidad física de la capa oculta 1 está representada en cada uno de los 13 instantes en el tiempo por una unidad virtual, con su activación específica. Todas las unidades virtuales que representan la misma unidad física tienen los mismos pesos entrantes. Por lo tanto, un TDNN puede considerarse como un MLP con algunos pesos de conexión limitados a ser iguales.

La prueba involucra varias épocas y se rastrea con el error cuadrático medio de las unidades de salida. Una iteración es la presentación de una sola muestra. Después del entrenamiento, la red se evalúa en las pruebas. Se reconoce un patrón cuando la unidad de salida con la activación real máxima corresponde a la unidad con la activación deseada igual a I. La tasa de reconocimiento se define como el porcentaje de muestras clasificadas correctamente. La tasa de error es, por lo tanto, el complemento de la tasa de reconocimiento con respecto a 100

Primero examinamos los efectos de modificar la función sigmoidea o la salida erro para obtener una superficie de error más pronunciada.

La presentación de un patrón modifica el peso conectando las unidades i a j, la tasa de aprendizaje de propagación hacia atrás es entonces proporcional a los valores de salida en la infinidad, hay varias formas de hacer que estas funciones no sean cero en el infinito.

Utilice un sigmoide simétrico cuyo valor nunca sea cero en el infinito, restando 0,5 del sigmoide. Esto generalmente proporciona una velocidad de aprendizaje ligeramente mejor, sin embargo, al comienzo de la fase de aprendizaje, cuando los pesos son pequeños y las activaciones cercanas a cero, el aprendizaje puede ser lento para iniciar.

**Estrategia de aprendizaje**

La estrategia de aprendizaje describe las partes de nuestro algoritmo de aprendizaje que se ocupan de la forma en que la muestra de entrenamiento se presenta a la red. Por ejemplo, ¿con qué frecuencia debemos actualizar los pesos y cuándo podemos omitir muestras? en comparación con los algoritmos de aprendizaje que actualizan el peso en cada época, aquellos que actualizan los pesos después de un pequeño número de iteraciones son mucho más rápidos y producen una mejor generalización. Sin embargo, actualizar los pesos después de cada iteración no parece útil y costoso de calcular, especialmente al final de la fase de aprendizaje, cuando uno tiene que sumar muchas iteraciones para obtener un valor delta significativo. Para nuestra tarea BDG de 3 clases, hemos utilizado el siguiente procedimiento: actualizar los pesos cada 3 iteraciones durante la primera época y luego incrementar el período de actualización en 3 en cada nueva época.

Escalado dinámico de los parámetros de aprendizaje.

Encontrar un valor óptimo para el tamaño del paso es un problema clave con la regla de aprendizaje de propagación hacia atrás. La mayoría de los métodos que se han propuesto para resolver este problema generalmente se refieren a la búsqueda de línea o algoritmos newton. Para nuestras tareas, traen dos tipos de problemas. Primero, generalmente asumen que los pesos se espacian, no intentan encontrar la solución con los pesos más pequeños posibles, como sucede con el aprendizaje muy lento. Por lo tanto, pueden producir un rendimiento de generalización más pobre