

Máquinas de Aprendizaje Multi-Tarea

Pedro José García Laencina, Jesús Serrano García, José Luis Sancho Gómez
Departamento de Tecnologías de la Información y las Comunicaciones
Universidad Politécnica de Cartagena. Plaza del Hospital 1, 30202 Cartagena
Teléfono: (+34) 968326542. E-mail: pedroj.garcia@upct.es

Resumen. La mayoría de máquinas de aprendizaje automático resuelven problemas con una sola tarea aislada, obviando las ventajas y el funcionamiento real del aprendizaje humano. Éste frecuentemente implica la resolución simultánea de distintas tareas; en particular, el aprendizaje de una tarea se ve mejorado y completado por el aprendizaje común con otras tareas relacionadas con la primera. Este artículo introduce los conceptos fundamentales del Aprendizaje Multi-Tarea (MTL, Multi-Task Learning) y presenta distintas arquitecturas neuronales para MTL. Los resultados obtenidos en un problema de clasificación de patrones muestran las ventajas de emplear máquinas MTL.

1. Introducción

El objetivo del aprendizaje automático es conseguir que una máquina (sistema computacional) sea capaz de utilizar datos o experiencias pasadas para resolver un problema que se le plantee. La máquina puede realizar automáticamente un aprendizaje que le permita solucionar problemas que requieren ciertas habilidades más allá de la mera capacidad de cálculo. Hoy en día existen numerosas aplicaciones reales del aprendizaje automático, incluyendo reconocimiento biométrico para identificación de personas, predicción de índices bursátiles, diagnóstico médica, etc.

Por contra, la mayor parte de estas aplicaciones son resueltas mediante el aprendizaje de una única tarea. Este aprendizaje (conocido como STL, *Single Task Learning*), obvia las ventajas y el funcionamiento real del aprendizaje humano. En éste, el aprendizaje de una tarea se ve mejorado y completado por el aprendizaje común con otras tareas relacionadas con la primera. Recientemente, muchos trabajos han aplicado estas ventajas al aprendizaje máquina [1]-[3]. Para ello durante el aprendizaje de una tarea principal se añaden tareas extra relacionadas con aquella. Este tipo de aprendizaje se conoce como *Aprendizaje Multi-Tarea* (MTL, *Multitask Learning*) [1].

El resto del artículo se estructura de la siguiente forma: en la Sección 2, se introducen los conceptos básicos y arquitecturas de las máquinas MTL. La Sección 3 muestra los resultados obtenidos en un problema de clasificación. Finalmente, se exponen las conclusiones principales.

2. Aprendizaje Multi-Tarea

Considérese una máquina STL tipo perceptrón multicapa (MLP, *Multi-Layer Perceptron* [4]) que consta de una capa oculta de neuronas completamente interconectadas y una salida asociada a una determinada tarea A (tarea principal) que se desea aprender (ver Fig. 1(a)). Esta máquina puede ser

entrenada (i.e., fase de aprendizaje) para resolver la tarea principal, a partir de un conjunto de vectores de entrada (patrones) asociados a dicha tarea [4].

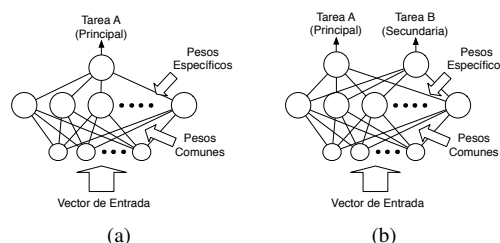


Fig. 1. Esquemas neuronales estándar para STL (a) y MTL (b).

Como se ha comentado anteriormente, es posible mejorar la capacidad de generalización de una máquina para aprender una determinada tarea (*tarea principal*) mediante el aprendizaje simultáneo con distintas tareas relacionadas (*tareas extra o secundarias*) [1], [2]. En una primera aproximación, un MLP multi-tarea se implementa añadiendo una salida extra para aprender una tarea secundaria B, junto con la principal, compartiendo todas las tareas una capa común de neuronas (ver Fig. 1(b)). El hecho de que un conjunto de neuronas estén conectadas a las salidas asociadas a las distintas tareas permite que lo que se aprenda en una salida contribuya al aprendizaje del resto, mejorando así la capacidad de generalización del MLP. Esta es la idea base del MTL: compartir la información aprendida mientras las tareas se aprenden en paralelo.

2.1. Subredes Privadas y Entradas Extra

Es posible mejorar el rendimiento y las prestaciones del MTL usando arquitecturas neuronales más complicadas que el esquema estándar MTL [3]. Para ello, se añade una subred específica o privada que aprende únicamente la tarea principal (Fig. 2(a)). Esta arquitectura presenta dos subredes: una *subred privada* empleada sólo para la tarea principal, y una *subred común* compartida por la tarea principal y las tareas secundarias. Este esquema es asimétrico ya

que la tarea principal afecta al aprendizaje de la tarea extra mediante la subred común; por el contrario, la tarea extra no puede afectar a la subred privada reservada para la tarea principal.

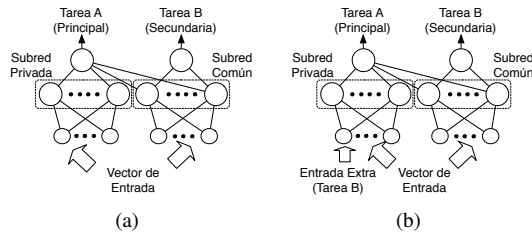


Fig. 2. Arquitecturas neuronales MTL con una subred privada.

Usando *entradas extra* es posible aumentar las prestaciones del MTL. Para ello, las salidas deseadas de la tarea B, junto con los vectores de entrada, se usan como entradas de la subred privada para aprender la tarea A (ver Fig. 2(b)). De esta forma, se añade información relacionada con la tarea principal en la subred privada, mejorando así el aprendizaje de la tarea principal. En el modo operación [4], es necesario usar el concepto de *consistencia* [3].

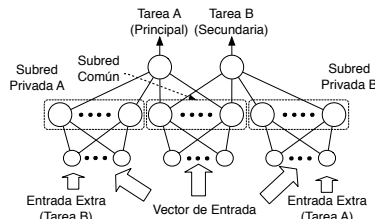


Fig. 3. Arquitectura neuronal MTL completa.

Finalmente, presentamos una última máquina MTL que emplea los conceptos de subredes privadas y entradas extra (ver Fig. 3). Esta arquitectura tiene una subred común que aprende todas las tareas, una subred privada para la tarea principal y otra subred privada asociada a la tarea extra. Además hace uso de entradas extra en cada subred privada. Esta máquina proporciona mejores resultados frente al resto de esquemas MTL propuestos anteriormente como se demuestra en la siguiente sección.

3. Resultados Obtenidos

Esta sección muestra los resultados obtenidos por las distintas arquitecturas presentadas para resolver el problema de clasificación de *Monk* [5]. Cada vector de entrada consta de seis características, i.e., $\mathbf{x} = [x_1, \dots, x_6]$, que están codificadas con valores enteros entre 1 y 4. El problema *Monk* consta de tres tareas, de las cuales en este trabajo consideraremos dos de ellas. En la tarea A (principal), se pretende identificar patrones con $(x_1 = x_2) \vee (x_5 = 1)$, mientras que en la tarea B (secundaria) se pretende identificar aquellos patrones donde al menos dos de sus seis atributos tomen el valor codificado como 1. El objetivo es mejorar el aprendizaje de la tarea principal usando esquemas MTL, por lo que se presentan únicamente

resultados relativos a dicha tarea, y no a la tarea secundaria. La Figura 4 muestra la evolución de la probabilidad de acierto para la tarea A en el conjunto de test a medida que avanza el entrenamiento. Cada curva es el promedio de 25 simulaciones. Podemos

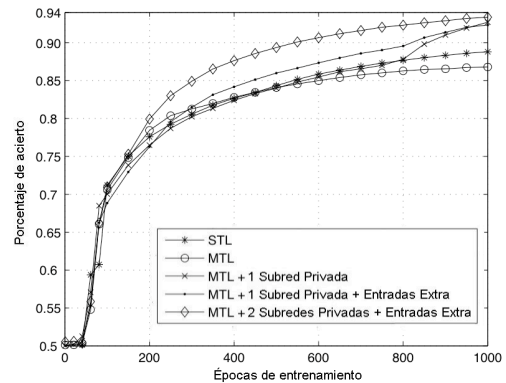


Fig. 4. Porcentaje de acierto para la tarea principal A (test).

observar como en el problema *Monk*, el esquema MTL estándar no proporciona mejores resultados que una solución STL. Pero por otro lado, se muestra el beneficio obtenido, tanto en términos de probabilidad de acierto como en términos de velocidad de convergencia, al añadir las subredes privadas y entradas extra al esquema MTL.

4. Conclusiones

Las máquinas MTL permiten entrenar una tarea principal junto con un conjunto de tareas extra relacionadas con la primera. Usando subredes privadas para cada tarea, y además entradas extra, las prestaciones del esquema estándar MTL son claramente mejoradas. Los resultados en un conocido problema de decisión son muestra de cómo el uso de máquinas MTL mejora la capacidad de generalización y la velocidad de convergencia de la tarea principal.

Agradecimientos

Este trabajo está parcialmente financiado por el Ministerio de Educación y Ciencia a través del proyecto TEC2006-13338/TCM y por la Fundación Séneca (Consejería de Educación y Cultura de Murcia) a través del proyecto 03122/PI/05.

Referencias

- [1] R. Caruana, *Multitask Learning*. Ph. D. Thesis, Carnegie Mellon University (1997).
- [2] D. Silver, *Selective Transfer of Neural Network Task Knowledge*, Ph. D. Thesis, Univ. of Western Ontario (2000).
- [3] P. J. García-Laencina, A. R. Figueiras-Vidal, J. Serrano, J.-L. Sancho-Gómez, "Exploiting Multitask Learning Schemes using Private Subnetworks". Proceedings of the 8th International Work-Conference on Artificial Neural Networks, pp. 233-240, Barcelona, Spain. 2005.
- [4] C. M. Bishop, *Neural Networks for Pattern Recognition*. New York, USA: Oxford University Press (1995).
- [5] S. Thrun *et al.*, "The MONK's Problem: a Performance Comparison of Different Learning Algorithms", Tech. Report, Carnegie Mellon University, 1991.