Garantía de calidad de servicio basada en la predicción del ancho de banda

Jesús Villadangos, Eduardo Magaña
Dpt. Automática y Computación, Universidad Pública de Navarra
Campus Arrosadía, 31006 - Pamplona - SPAIN
Teléfono. : +34 948 169645 Fax: +34 948 168924
E-mail: jesusv@unavarra.es

Abstract*. This paper presents the architecture and performance evaluation of a neuronal estimator to predict network load in communication networks. System benchmarks are tested with real network traffic picked up from the 155 Mbps ATM Internet connection of the Universidad Pública de Navarra. The system shows good benefits in traffic prediction with 3 and 5 hours of advance. So the system shows characteristic of great interest to carry out the dynamic assignment of bandwidth in Internet Service Providers (ISPs), guaranteeing quality of service hired by the users.*

1 Introducción

Los usuarios residenciales acceden a las redes de datos a través de un proveedor de servicios (ISP). Actualmente, cada ISP tiene contratado un acceso a una operadora de red de datos y sobre dicho acceso multiplexa a sus clientes, por lo que el ancho de banda se comparte por los usuarios.

Un parámetro de diseño para los operadores de un ISP es, por tanto, el número de usuarios a los que se va a proporcionar servicios. Sin embargo, más importante es conocer qué tipo de servicios van a ser usados por sus clientes y, en concreto, el consumo de ancho de banda que requiere cada usuario. El estudio del ancho de banda requerido por el ISP para dar servicio a sus clientes permite determinar la tasa que éste contratará a la operadora de red. La selección de un ancho de banda es un compromiso entre el coste del acceso y el grado de servicio que se quiere proporcionar a los usuarios. A los ISPs les interesa maximizar el uso del ancho de banda contratado.

El objeto principal de este trabajo es presentar una arquitectura de estimador del ancho de banda basada en una red neuronal que permite predecir el ancho de banda requerido para asegurar el servicio a sus clientes. Se presenta los resultados de su aplicación para el caso del enlace de entrada y salida de la Universidad Pública de Navarra.

El sistema se basa en el uso de un estimador de la carga de la red que permite predecir con horas de anticipación los bytes totales transferidos por los usuarios. El estimador consiste en un filtro FIR realizado mediante una red neuronal [1]. Las redes neuronales se han aplicado en muchos casos para la identificación y control de sistemas [2, 3] y de sistemas de telecomunicación (codificación de datos, control de errores, control de admisión de llamadas, control de congestión, etc.) [4, 5, 6]. Sin embargo, en todos los casos se han utilizado siempre modelos de tráfico sintético y no datos reales de tráfico.

Los datos para entrenar la red neuronal se obtienen de la captura de todos los paquetes que circulan por el punto de acceso (TR1) de la Universidad Pública de Navarra con RedIris, la troncal que une a las universidades españolas con el resto de Internet. El tráfico analizado se puede considerar representativo del tráfico que maneja un ISP, ya que la universidad tiene un alto número de usuarios que hacen uso de muy diversos servicios de Internet.

En este trabajo se presenta, por tanto, la adecuación de un estimador neuronal para la predicción de la carga de la red de la Universidad. Se analiza tanto el canal de bajada como el de subida haciéndose la predicción con una anticipación entre una y cinco horas.

En primer lugar se presentará la arquitectura de red neuronal que permite realizar predicciones sobre series temporales. Esta propuesta se centra en los resultados que se han obtenido al hacer uso de una red neuronal para estimar el tráfico real de la Universidad Pública de Navarra. A continuación se analiza la carga de la red para diferentes niveles de predicción. Finalmente se presentan las conclusiones y referencias del trabajo.

2 Arquitectura del estimador neuronal

Es bien conocido que las redes neuronales son capaces de realizar mapeos no lineales entre conjuntos de entradas y salidas. Una red neuronal feedforward de tres capas con neuronas, cuya función de activación es de tipo sigmoidal es capaz de aproximar una función no lineal con cualquier

grado de precisión [1]. Sin embargo, este tipo de redes no están diseñadas para tener en cuenta la dinámica de las señales variables en el tiempo.

Un método para representar el tiempo en las redes neuronales es utilizar una red de tipo Time Delay Neural Network (TDNN), la cual es una red multicapa feedforward en la que las salidas de las neuronas se almacenan durante un número finito de intervalos de tiempo y sirven de entrada para las neuronas de la siguiente capa.

La topología de las redes TDNN está incluida en las redes perceptron multicapa considerando que cada una de las salidas en realidad es la respuesta de un filtro FIR (*Finite Impulse Response*), donde FIR indica que para una entrada de duración finita, la salida del filtro tiene una duración finita. Este tipo de redes se denominan perceptrones multicapa FIR. Ambas redes TDNN y FIR son funcionalmente equivalentes. Sin embargo, la red FIR está directamente relacionada con las redes multicapa. Las redes FIR, además, permiten derivar esquemas de adaptación de modo sencillo. Por tanto, en este trabajo se hará uso de una red FIR como sistema de predicción del tráfico.

2.1 Modelo de red FIR

Como se ha indicado anteriormente, las redes multicapa permiten realizar un mapeo estático. A estas redes se les aplica una modificación transformando el peso de cada una de las conexiones por un filtro FIR lineal. Para este filtro, la salida *y(k)* se corresponde con la suma ponderada de los valores de entradas de instantes anteriores:

$$y(k)=\sum_{n=0}^{T} w(n)x(k-n) \tag{1}$$

A partir de la ecuación (1) se puede formular el modelo de una neurona FIR. Sea $w_{ij}(l)$ el peso que corresponde al filtro FIR de la conexión que une la neurona *i* con la neurona *j* (*i*=1,2…,p). El parámetro *l* toma valor en el intervalo [0, M], donde M es el número total de unidades de retardo que se consideran al diseñar el filtro FIR. Finalmente, sea $y_j(n)$ el valor de la función de salida de la neurona *j* y $x_i(n)$ la señal de entrada. Entonces, se tiene que

$$v_j(n)=\sum_{i=1}^{p}\sum_{l=0}^{M} w_{ji}(l)\, x_i(n-l)-\theta_j \tag{2}$$

$$y_j(n)=\varphi(v_j(n)) \tag{3}$$

donde $v_j(n)$ es el potencial de activación de la neurona *j*, θ_j es el umbral externo para la neurona *j* y $\varphi(.)$ es la función no lineal de activación de la neurona.

Las ecuaciones 2 y 3 se pueden reformular en forma matricial, donde se va a hacer uso de las siguientes definiciones para el vector de estado y el vector de pesos para la conexión *i*, respectivamente:

$$\chi_i(n)=[x_i(n), x_i(n-1),..., x_i(n-M)]^T \tag{4}$$

$$\varpi_{ij}=[w_{ji}(0), w_{ji}(1),..., w_{ji}(M),]^T \tag{5}$$

El valor de salida $y_j(n)$ de la neurona *j* se expresa del siguiente modo:

$$y_j(n)=\varphi\left(\sum_{i=1}^{p}\varpi_{ji}^T\,\chi_i(n)-\theta_j\right) \tag{6}$$

Además, el modelo de la red FIR se basa en neuronas cuya estructura considera un peso w_{0j} conectado a la entrada fija x_0 = -1 para representar el umbral externo θ_j.

A partir del modelo de neurona anterior se puede construir una red de tipo perceptrón multicapa cuyas neuronas ocultas y de salida se basan en el modelo de filtro FIR. Esta estructura de red se denomina red FIR. La diferencia entre las redes FIR y las redes multicapa tradicionales reside en que las conexiones estáticas de las redes multicapa se cambian por versiones dinámicas. En redes multicapa tradicionales la conexión entre dos neuronas está ponderada por un peso, mientras que en las redes FIR este peso se transforma en un conjunto de pesos asociados cada uno de ellos a la entrada en instantes anteriores.

2.2 Aprendizaje backpropagation temporal

Dada una secuencia de entrada *x(k)*, la red produce una secuencia de salida *y(k) = N(W, x(k))*, donde *W* representa el conjunto de todos los coeficientes de los filtros presentes en la red. Se define el error instantáneo $e^2(k) = ||d(k) - y(k)||^2$ como la distancia euclídea entre la salida de la red y la salida deseada. Por tanto, el objetivo del entrenamiento se corresponde con ajustar el valor de los coeficientes de *W* para minimizar la siguiente función de coste:

$$C=\frac{1}{2}\sum_{k=1}^{K} e^2(k) \tag{7}$$

donde la suma se realiza sobre el conjunto total *K* de muestras de aprendizaje. El algoritmo para minimizar el error de la función *C* se presenta en [7] y se denomina backpropagation con tiempo. La función de adaptación de los pesos se presenta a continuación:

$$\varpi_{ji}(k+1)=\varpi_{ji}(k+1)-\eta\frac{\partial C}{\partial v_j(k)}\frac{\partial v_j(k)}{\partial \varpi_{ji}(k)}=\varpi_{ji}(k+1)-\eta\delta_j(k)\chi_i(k) \tag{8}$$

$$\delta_j(k)=\begin{cases} e_j(k)\varphi'(v_j(k)), & capa\,de\,salida \\ \varphi'(v_j(k))\sum_{m\in A}\Delta_m^T(k)\varpi_{mj}, & capa\,oculta \end{cases} \tag{9}$$

donde η es el parámetro de velocidad de aprendizaje, A se define como el conjunto de todas las neuronas cuyas entradas se ven afectadas por el valor de salida del nodo *j* y $\Delta_m(k)$ se define del siguiente modo:

$$\Delta_m(k)=[\delta_m(k),\delta_m(k+1),...,\delta_m(k+M)]^T \quad (10)$$

El conjunto de ecuaciones anteriores representa una generalización del algoritmo de aprendizaje clásico de backpropagation. De hecho, se puede reemplazar el vector de entrada $\chi_i(n)$, el vector de pesos $\varpi_{mj}(n)$, y el vector gradiente Δ_m por sus correspondientes valores escalares y se obtendría el algoritmo backpropagation para redes estáticas. Para calcular el valor $\delta_j(k)$ para una neurona *j* de una capa oculta, se filtra el resultado de las δ de las siguientes capas hacia atrás a partir del conjunto de nodos que se ven afectados por el valor de salida de la neurona *j*. Por tanto, el valor de las δ no está afectado sólo por los valores de los pesos, sino que se adapta en función del resultado del filtrado de la señal hacia atrás. Para cada nueva entrada y respuesta deseada, los filtros hacia delante y hacia atrás se incrementan una unidad temporal. Entonces, los pesos se adaptan on-line para cada intervalo de tiempo.

Utilizar el algoritmo backpropagation con tiempo hace que se preserve la simetría entre la propagación hacia delante de los estados y la propagación hacia atrás de los errores. Se mantiene por tanto el procesamiento paralelo en el sistema. Además, cada peso se utiliza de forma individual y una sola vez para calcular el valor de las δ; es decir, no hay redundancia en el instante de aplicar el modelo del gradiente.

Sin embargo, un análisis detallado de las ecuaciones anteriores permite mostrar que no se tiene un orden causal a la hora de determinar los valores de $\delta_j(k)$. Este cálculo se puede expresar de forma causal el algoritmo backpropagation con tiempo del siguiente modo:

Para cada neurona *j* de la capa de salida, calcular

$$\varpi_{ji}(k+1)=\varpi_{ji}(k)+\eta\,\delta_j(k)\,\chi_i(k) \quad (11)$$

$$\delta_j(k)=e_j(k)\varphi'_i(k) \quad (12)$$

Para cada neurona *j* de una capa oculta, calcular

$$\varpi_{ji}(k+1)=\varpi_{ji}(k)+\eta\,\delta_j(k-lM)\chi_i(k-lM) \quad (13)$$

$$\delta_j(k-lM)=\varphi'(v_j(k-lM))\sum_{m\in\Lambda}\Delta_m^T(k-lM)\,w_{mj} \quad (14)$$

donde *M* es la longitud del filtro y se usa el índice *l* para identificar la capa oculta. Es decir, *l* = 1 se refiere a la primera capa oculta anterior a la capa de salida.

3 Estimación de la carga usando redes FIR

Las redes neuronales tienen capacidad de adaptación y permiten modelar la ausencia de estacionariedad de los sistemas. Sus capacidades de generalización las hacen herramientas flexibles y robustas cuando se está tratando con datos que incluyen patrones ruidosos. En este trabajo, el papel de la red neuronal es capturar la compleja relación entre valores de carga pasados y futuros. El objetivo es predecir cual va a ser la carga del enlace en instantes futuros con el fin de proponer un sistema que permita prever los recursos de comunicaciones necesarios para el sistema.

3.1 Configuración del estimador durante el aprendizaje

Sea *x(k)* una serie temporal escalar, la cual se describe por un modelo de regresión no lineal de orden q como sigue:

$$x(n)=f(x(n-1), x(n-2), \ldots, x(n-q)) + \varepsilon(n) \quad (15)$$

donde *f* es una función no lineal de sus argumentos y $\varepsilon(n)$ es un residuo. Se asume que $\varepsilon(n)$ se puede representar por un ruido blanco gausiano. La funcion no lineal *f* se desconoce a priori, y la única información que se posee es la observado y representada por la serie *x(1)*, … , *x(N)*. *N* determina el número total de muestras de la serie. Se puede usar una red FIR como estimador de un paso y orden q para modelar la serie temporal. De hecho la red se diseña para realizar la estimación del valor *x(n)* teniendo en cuenta las q entradas pasadas. Es decir, *x(n-1)*, …, *x(n-q)*, como se indica en la expresión del estimador:

$$\hat{x}(n)=F(x(n-1),...,x(n-q))+e(n) \quad (16)$$

En este trabajo se utiliza el estimador para predecir con diferentes pasos.

La función no lineal *F* es una aproximación de la función f, la cual se calcula mediante la red FIR. El valor *x(n)* actúa como valor deseado. Así, la red FIR se entrena con el objetivo de minimizar el error de predicción:

$$e(n)=x(n)-\hat{x}(n) \quad q+1\le n\le N \quad (17)$$

En nuestro caso, la red FIR se diseña como una red totalmente conectada con tres capas de 1, 8 y 1 neuronas en cada capa a partir de la de entrada y con 3 elementos para llevar a cabo los filtros FIR. La selección se ha realizado mediante el método de prueba/error, ya que no se conocen en la literatura métodos que determinen la configuración de las redes neuronales. La red FIR se entrena utilizando la forma causal del algoritmo backpropagation temporal y se usa el error cuadrático medio como la

medida del error. El parámetro de aprendizaje se establece a valor 0.1 y se utiliza en cada neurona como función de activación la sigmoidal.

3.2 Modelo de tráfico

El entrenamiento y validación del estimador se realiza mediante el uso de una traza de tráfico capturada en la Universidad Pública de Navarra desde el 27 de noviembre de 1998 al 11 de enero de 1999 de la que se obtienen los bytes por hora en el enlace de la universidad tanto en la bajada (downstream), como en la subida (upstream). Se distinguen los dos casos debido a la asimetría que presenta el tráfico en la red, siendo el tráfico de bajada mayor que el de subida debido a que los usuarios demandan del exterior más información que la proporcionada por servidores de la universidad a usuarios externos. La traza de tráfico contiene información sobre 1095 horas (46 días) siendo los valores de cada hora los bytes transferidos en cada sentido durante cada hora.

4 Análisis de prestaciones del estimador de la carga de tráfico

Este apartado muestra las prestaciones del estimador en dos casos: (i) estimación del ancho de banda requerido por los usuarios en la hora siguiente y (ii) la predicción para dentro de tres y cinco horas a partir de la actual. En el primer caso se estudia las posibilidades de estimación usando como información de partida el día de la semana y la hora del día. A continuación se basa la predicción en el uso del valor de la carga para horas anteriores junto con el día de la semana. En el segundo caso se utilizan las entradas retrasadas y el día de la semana para determinar el ancho de banda requerido por los usuarios con tres y cinco horas de antelación.

4.1 Estimación del ancho de banda para la hora siguiente

En este apartado se analiza la estimación obtenida por la red neuronal para los enlaces de bajada y subida considerando que se tiene como información de partida (i) en primer lugar, el día de la semana y la hora del día y (ii) en segundo lugar, el día de la semana y el valor del ancho de banda requerido por los usuarios en X horas anteriores.

En el primer caso, en cuanto al canal de bajada, la estimación de la carga del enlace se muestra en la Fig.1, mientras que la Fig.2 muestra dicha estimación para el enlace de subida. En ambos casos se puede comprobar que la red neuronal determina el comportamiento del sistema como periódico.

En este caso se comprueba que las entradas no son suficientemente generales para poder modelar el comportamiento de las necesidades de ancho de banda de los usuarios. El estimador, sin embargo, ha generalizado el comportamiento de los usuarios a un esquema de alta demanda durante los días laborables y de baja demanda durante los días no laborables.

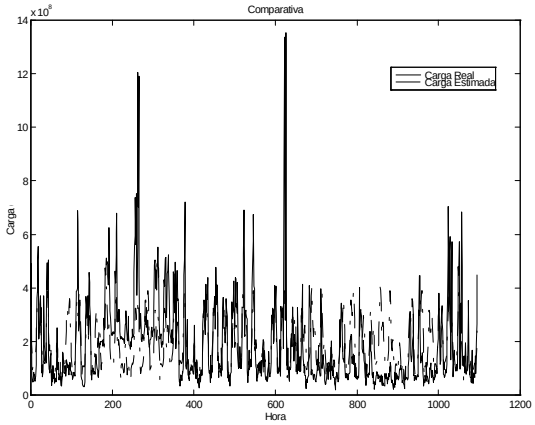

Figura 1 : Tráfico en el canal de bajada considerando día de la semana y hora del día.

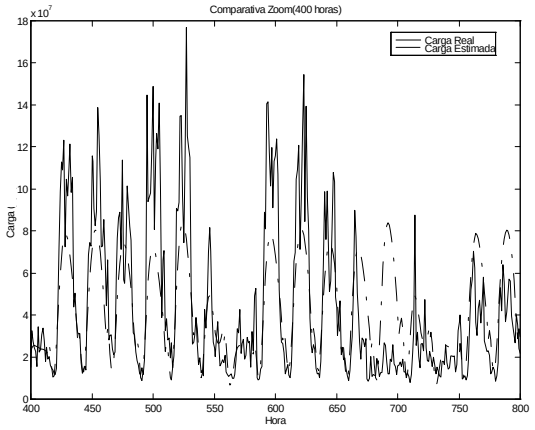

Figura 2 : Tráfico en el canal de subida considerando día de la semana y hora del día.

En segundo lugar se estima el ancho de banda requerido por los usuarios en la siguiente hora pero usando como información de entrada el día de la semana, la hora del día y el ancho de banda requerido en X = 5 horas anteriores.

La Fig. 3 muestra la predicción para el canal de bajada. En este caso se puede comprobar como el estimador ha sido capaz de generalizar los datos de aprendizaje y permite una predicción bastante buena para la hora siguiente.

A continuación se muestra en la Fig. 4 un detalle de la estimación superponiendo la carga real y la estimada. En este caso se puede comprobar que la realización del estimador es bastante aproximada a la carga real.

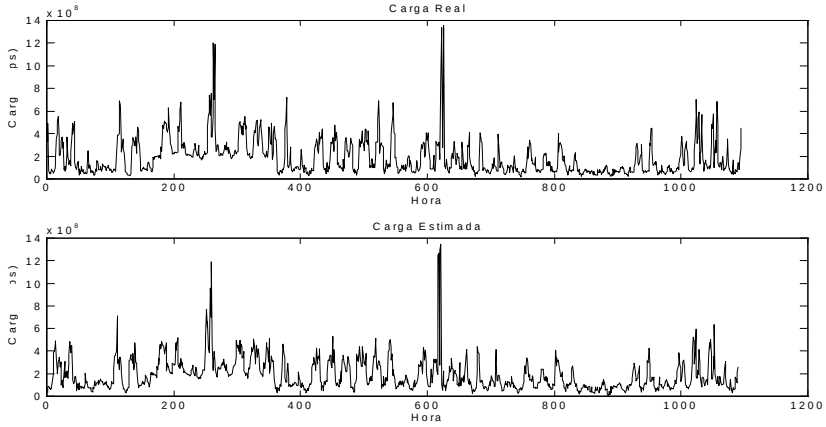

Figura 3: Estimación del ancho de banda del canal de bajada considerando el día de la semana, la hora del día y el valor de la carga en cinco horas anteriores.

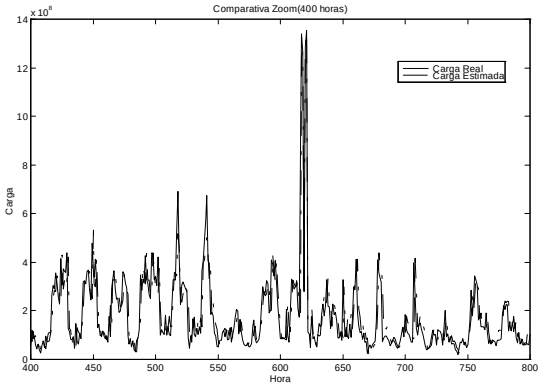

Figura 4 : Detalle de la estimación del ancho de banda en el canal de bajada considerando el día de la semana, la hora del día y el valor de la carga en cinco horas anteriores.

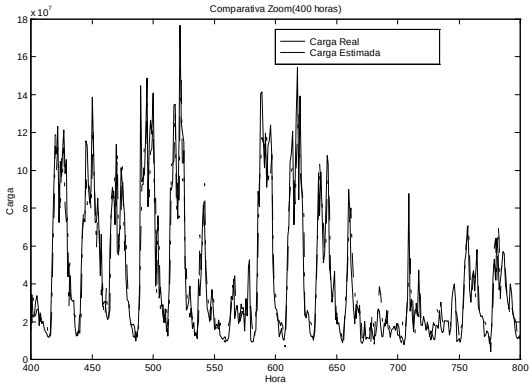

Figura 6 : Detalle de la estimación del ancho de banda en el canal de bajada considerando el día de la semana, la hora del día y el valor de la carga en cinco horas anteriores.

La Fig. 5 muestra la predicción para el canal de subida, así como la Fig. 6 muestra un detalle para dicha predicción. En este caso, como en el anterior para el canal de bajada se puede comprobar que el estimador puede ser de gran utilidad para un proveedor de servicios de Internet para determinar el ancho de banda que van a requerir los usuarios en la siguiente hora.

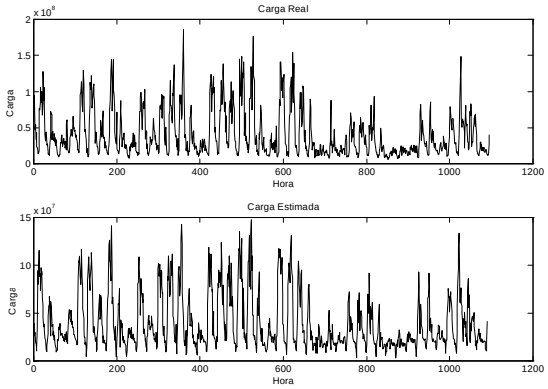

Figura 5 : Estimación del ancho de banda en el canal de subida considerando el día de la semana, la hora del día y el valor de la carga en cinco horas anteriores.

En este apartado se ha mostrado que el uso de variables como el día de la semana, la hora del día y entradas anteriores permiten determinar el ancho de banda requerido por los usuarios en la hora siguiente. Sin embargo, para un proveedor de servicios resulta de mayor interés estimar con precisión el ancho de banda requerido con antelación y cuanta mayor antelación mejor. Esto permitiría tomar acciones preventivas al proveedor de servicios para disponer del ancho de banda en caso de ser necesario aumentar el contratado o bien reducir el ancho de banda contratado si no se va a utilizar.

4.2 Predicción del ancho de banda requerido con horas de antelación

El estimador trabaja en este caso con los siguientes datos: día de la semana, hora del día y valor del ancho de banda en X=5 horas anteriores. El resultado obtenido es el valor de ancho de banda que se predice con Y= 2 horas de antelación.

La Fig.7 muestra la capacidad de predicción del sistema para el caso del canal de bajada y la Fig. 8 muestra la capacidad de predicción del sistema para el canal de subida. En ambos casos se puede comprobar que la red neuronal realiza una predicción que no se ajusta exactamente a la carga real pero que puede ser una buena aproximación para determinar el ancho de banda que requiere el proveedor de servicios para dar servicio a sus usuarios.

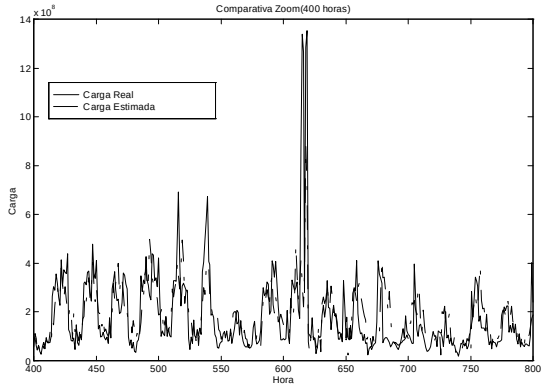

Figura 7 : Detalle de la estimación del ancho de banda en el canal de bajada con dos horas de antelación considerando el día de la semana, la hora del día y el valor de la carga en cinco horas anteriores.

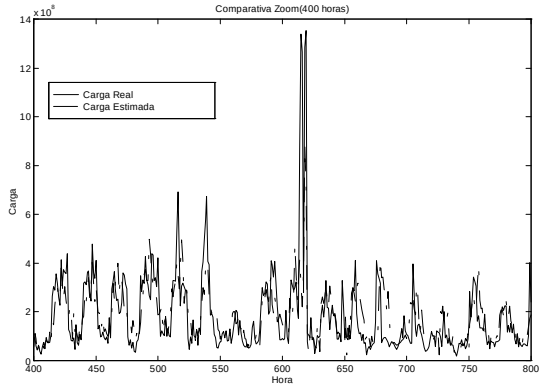

Figura 8 : Detalle de la estimación del ancho de banda en el canal de subida con dos horas de antelación considerando el día de la semana, la hora del día y el valor de la carga en cinco horas anteriores.

Conclusiones

El presente trabajo muestra la utilidad de las redes neuronales como estimadores de tráfico en redes de datos, las cuales se caracterizan por tener una alta variabilidad en el patrón del tráfico que circula por ellas. La estimación del ancho de banda permite prever con suficiente antelación el ancho de banda que debe contratar el proveedor de servicios para dar servicio a sus usuarios. Esto permite que los proveedores puedan contratar un ancho de banda básico y activar líneas de respaldo cuando la demanda de ancho de banda se prevea muy alta. Esto permitiría reducir costes de contratación a los proveedores de servicios.

En este trabajo, además, se muestra la posibilidad de integrar de manera natural en el estimador entradas de gran contenido semántico como son el día de la semana y la hora del día además de los valores del ancho de banda requerido por los usuarios en horas anteriores. Estas entradas no se pueden integrar en la realización de estimadores clásicos y, sin embargo, son de gran utilidad para la realización de las previsiones de un operador de red porque tienen en cuenta datos importantes como la estacionalidad del tráfico.

La estimaciones realizadas con el sistema propuesto se han realizado sobre tráfico real mostrando la bondad del sistema y las posibilidades que proporciona para un proveedor de servicios. Las redes neuronales muestran su capacidad para predecir el tráfico que circula por un enlace de comunicaciones, aunque se debe buscar una arquitectura de red u otro tipo de sistemas para ajustar la estimación a la carga real del sistema e intentar la predicción con mayor antelación.

Referencias

[1] Z. Fan, P. Mars; *ATM traffic prediction using FIR neural networks.* ATM Networks: Performance modelling and evaluation, vol II. Chapman & Hall, 1996

[2] B. Kosko; *Neural networks and fuzzy systems: A dynamical systems approach to machine intelligence*. Prentice Hall, 1991.

[3] J. E. Neves, M. J. Leitao, L. B. Almeida; *Neural networks in B-ISDN flow control: ATM traffic prediction or Network modeling?*. IEEE Communications Magazine, Oct. 1995.

[4] A. Hiramatsu; *ATM communications network control by neural networks.* IEEE Transactions on Neural Networks, v. 1, n. 1, March 1990.

[5] Y-K. Park, G. Lee; *Applications of neural networks in high-speed communication networks.* IEEE Communications Magazine, Oct. 1995.

[6] R-G. Cheng, C-J. Chang, L-F. Lin*; A QoS-provisioning neural fuzzy connection admission controller for multimedia high-speed networks*. IEEE/ACM Transactions on Networking, v. 7, n. 1, Feb. 1999.

[7] E. A. Wan; *Time series prediction by using a connectionist network with internal delay lines*. In Time Series Prediction, 195-217, Addison-Wesley.