Expert Systems with Applications xxx (2015) xxx–xxx

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Comparing forecasting approaches for Internet traffic

Christos Katris *, Sophia Daskalaki

Department of Electrical & Computer Engineering, University of Patras, Rio 26504, Greece

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Internet traffic
Long memory time series
Nonlinear time series
FARIMA models
Neural networks
Hybrid models

ABSTRACT

In this paper, we experiment with several different forecasting approaches for Internet traffic and a scheme for their evaluation. First the existence of properties such as Short or Long Range Dependence and non-linearity is explored in order to take advantage of such information and offer a couple of alternatives as forecasting models. The proposed models include FARIMA with Normal and Student's t innovations and two different architectures of Artificial Neural Networks, the Multilayer Perceptron and Radial basis function. Next, we construct a model selection scheme based on the White's Neural Network test for non-linearity or alternatively combine FARIMA and Neural Networks into hybrid forecasting models. The comparison of all suggested approaches is performed using their average position and standard deviation of position when applied to several known datasets of Internet traffic and when the accuracy of forecasts is measured with three different measures. Based on such a data analysis it is shown that hybridization and the selection of a model according to a non-linearity test are more successful as forecasting approaches over all individual models, as well as over other well-known methods such as Holt–Winters, ARIMA/GARCH and FARIMA/GARCH. This result indicates that forecasting approaches which take non-linearity into account lead to better overall forecasts for Internet traffic.

1. Introduction

Forecasting of Internet traffic is very important for tasks such as resource allocation, network planning and detection of network anomalies caused by attacks. This is the case since improved forecasting of TCP/IP traffic can help network providers optimize their resources. In bandwidth allocation schemes, better traffic forecasts can help obviating congestion or waste of resources. Moreover, an accurate prediction model can be used to detect security attacks in computer networks, by comparing predicted with actual traffic.

Internet traffic, as it is well known, carries the properties of statistical self-similarity and Long Range Dependence (LRD) (Beran, Sherman, Taqqu, & Willinger, 1995; Dymora, Mazurek, & Strzalka, 2013; Leland, Taqqu, Willinger, & Wilson, 1994). While these properties can be detected in almost all traces from Internet, this is not the case with the property of non-linearity. As it is shown in this article, in some cases but certainly not always, nonlinear terms may be needed in the functional form of the model that describes the time series. Especially when forecasting or performance modeling is the ultimate goal, detecting such properties in traces is important since they influence the choice of the model to be adopted.

Classical forecasting models such as the Autoregressive (AR) and the Autoregressive Integrated Moving Average (ARIMA) can capture the linear and Short Range dependencies (SRD) between terms, but not LRD, thus often resulting to poor performance when used for Internet traffic forecasting. Nevertheless their use is extensive. For example, in Zhani, Elbiase, and Farouk (2009) Ethernet traffic, in Won and Ahn (2005) video traffic, in Rutka (2009) traffic from a website, in Moussas, Daglis, and Kolega (2005) data from a campus network and in Takahashi, Aida, and Saito (2000) from a university router, are all modeled using such models. To overcome the weaknesses of ARIMA, self-similar models such as Fractional Gaussian Noise (Mandelbrot & Van Ness, 1968) and FARIMA $(0,d,0)$ have been proposed instead. More recent measurements on Internet traffic, however, unveiled the co-existence of LRD and SRD (Shu et al., 1999). Therefore it appears that the most appropriate category is the Fractional ARIMA (FARIMA) (p,d,q) (Granger & Joyeux, 1980; Hosking, 1981), which can describe both types of dependencies. In fact, various types of Internet traffic have been modeled with the help of such models. For example, in Corradi, Garroppo, Giordano, and Pagano (2001) FARIMA was used to model Ethernet and Video, in Shu et al. (1999) for Ethernet traffic, in Chiruvolu and Sankar (1997) for video traffic and in Dethe and Wakde (2004) for LAN and WAN

* Corresponding author.
E-mail addresses: chriskatris@upatras.gr (C. Katris), sdask@upatras.gr (S. Daskalaki).

http://dx.doi.org/10.1016/j.eswa.2015.06.029
0957-4174/© 2015 Published by Elsevier Ltd.

traffic. The main difficulty with FARIMA models, however, is the estimation of the fractional difference parameter d. One of the many approaches, developed for this purpose, is the Geweke–Porter–Hudak (GPH) estimator (Geweke & Porter-Hudak, 1983).

Forecasting models based on Artificial Neural Networks (ANN) have also been proposed for many different disciplines and Internet traffic cannot be an exception to that. Their strength is exposed mainly when the time series carries nonlinear terms. Time series prediction using ANNs can be found in Balkin and Ord (2000), Dorffner (1996), Frank, Davey, and Hunt (2001), Patterson, Chan, and Tan (1993). Specifically for Internet traffic prediction ANNs have been employed in Alarcon-Aquino and Barria (2006), Cortez, Rio, Rocha, and Sousa (2012), Edwards, Tansley, Davey, and Frank (1997), Jiang and Papavassiliou (2004), Katris and Daskalaki (2014), Wang, Zhang, Yan, and Zheng (2008). Moreover, time series prediction with Radial Basis Function (RBF) neural networks can be found in Gowrishankar and Satyanarayana (2009), Ma and Xu (2007), Szmit, Szmit, and Kuzia (2013).

In this paper, we experiment with a number of different forecasting approaches all of which are suitable for time series carrying the properties of Internet traffic. More specifically, we build FARIMA and ANN forecasting models based on traces from the Internet. In order to widen our experimentation with the different ANN models, both Multilayer Perceptron (MLP) and RBF architectures have been used, while for the FARIMA models both Normal and Student's t innovations are considered. We present all necessary details for their model construction procedures and then develop two alternative procedures of combining them so to take advantage of their synergy. The first is based on the White Neural Network (WNN) test (Lee, White, & Granger, 1993), a statistical test for non-linearity, which can be performed on the dataset in hand and let us decide whether the assumption of linearity is significant. The test can then be used as a guide for choosing between the FARIMA and the ANN models. The second approach refers to hybrid models where FARIMA and ANN models are combined constructively. Following a hybridization technique where initially a FARIMA model is fit on the data and afterwards the residuals are modeled with the help of Neural Networks (Aladag, Egrioglou, & Cadilar, 2012; Zhang, 2003) we take advantage of the strengths of both approaches. The two forecasting approaches are compared with the individual FARIMA and ANN models using the Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) as performance metrics. Since the approaches to be compared are many, however, we take into account the average position and standard deviation of every model in a set of traces. By doing this each model is graded based on its capacity to forecast systematically better than the others. In addition, the two approaches are compared with other well-known alternatives for Internet traffic forecasting such as ARIMA/GARCH and Holt–Winters, but also with FARIMA/GARCH process which also captures LRD and non-linearity.

The remaining of this paper is as follows. Section 2 reviews the properties of Internet traffic and discusses the concepts of long-memory and non-linearity. Section 3 discusses analytical procedures for the construction of FARIMA and ANN models for Internet traffic traces, as well as the building procedure for the hybrid FARIMA–ANN model. In addition, it goes over the WNN Neural Network test for non-linearity and its incorporation into a selection procedure between a FARIMA and an ANN model for a given trace. Section 4 presents a framework for the evaluation and comparison of different models, and Section 5 gives the results of an extended experimentation with several Ethernet and video datasets, used to test the forecasting capability of the different models and approaches previously discussed. Lastly, Section 6 summarizes and concludes the presentation of this work.

2. Internet traffic and its properties

According to previous studies, traces from Internet traffic carry the properties of SRD, LRD and sometimes non-linearity. The concept of LRD in time series is related to the decay of the autocorrelation function in Eq. (1), which in the case of long memory is slower than exponential. For a stationary stochastic process $\{X_t\}$ the autocorrelation function is defined as

$$\rho(k) = \frac{E[(X_t - \mu)(X_{t+k} - \mu)]}{\sigma^2} \qquad (1)$$

where μ and σ are the mean and standard deviation of X_t, respectively. The autocorrelation function gives a measure of similarity between the processes X_t and X_{t+k}, for all k, and so offers a way for characterizing the dependence that the time series holds. If the autocorrelation function $\rho(k)$ decays to zero with an exponential rate, then the stochastic process exhibits short range dependence. On the contrary, if it decays to zero hyperbolically, i.e. so slowly that $\sum_{k=1}^{\infty}\rho(k) = \infty$, then X_t exhibits long memory. In practice this implies that for large values of k, the autocorrelation function has approximately a power-law shape $\rho(k) = k^{-\beta},\ 0 < \beta < 1$. The presence of correlation even after a very large number of lags attributes quite special properties to the process; specifically, there is no timescale where the assumption of independence may hold even approximately.

The main measure for long memory in time series is the Hurst exponent (H), which was first introduced in the field of hydrology (Hurst, 1951), and since then it has been used in a variety of fields. If $0.5 < H < 1$, then we know that the series exhibits long memory and the closer it is to 1, the stronger the long memory. On the contrary, the value $H = 0.5$ indicates absence of long memory, i.e. uncorrelated series or series with autocorrelation function that decays exponentially to zero. If $0 < H < 0.5$ antipersistence is indicated, that is high values are more likely to be followed by low values and vice versa. Lastly if $H > 1$, then the only possible conclusion one can derive is non-stationarity for the series. Estimation of the Hurst exponent can be achieved by several methods; however, for our experimental work the R/S method (Hurst, 1951; Mandelbrot, 1972) was used. Details for this estimation method can be found in (Peters, 1994).

Non-linearity is a characteristic that may exist in some Internet traces and not in others. On the other hand, suspicion for non-linear terms in the functional expression of the dependence in a time series is one of the reasons for using ANNs in forecasting instead of the classical ARIMA or FARIMA models. Therefore a statistical test that can be used to test whether the nature of data is linear or not, is clearly very important. The WNN test for non-linearity (Lee et al., 1993) is one such statistical test, developed on the basis of a neural network structure, and can check the significance of the linearity assumption. The null hypothesis of the test is linearity in mean, implying that a linear model is suitable to describe the data, while the alternative hypothesis supports the assumption of an arbitrary structure.

3. Forecasting approaches for Internet traffic

Given the special characteristics of Internet traffic a number of alternative forecasting procedures may be proposed, including FARIMA, ARIMA or FARIMA with GARCH and ANNs. In this section we briefly present procedures for the construction of FARIMA models with Normal or Student's t innovations and Neural Networks with MLP or RBF architectures. Given the models that can be constructed, we go over the WNN non-linearity test and describe a procedure that allows selecting in advance between a FARIMA and an ANN model. Lastly, an alternative to the selective procedure

Please cite this article in press as: Katris, C., & Daskalaki, S. Comparing forecasting approaches for Internet traffic. *Expert Systems with Applications* (2015), http://dx.doi.org/10.1016/j.eswa.2015.06.029

is a hybrid model which is described in detail and attempts to capture SRD, LRD and non-linearity (if it exists) concurrently.

3.1. FARIMA Models

Like ARIMA, these models are also linear in mean with the additional feature that can handle the presence of LRD and SRD in the data; therefore are suitable for the modeling of Internet traffic. A FARIMA time series model is formulated as in Eq. (2):

$\Phi_p(L)(1-L)^d(X_t) = \Theta_q(L)\varepsilon_t$, where L is the lag operator, $\Phi_p(L) = 1 - \varphi_1 L - \ldots - \varphi_p L^p$,

$$\Theta_q(L) = 1 + \theta_1 L + \ldots + \theta_q L^q \text{ and } (1-L)^d = \sum_{j=0}^{\infty} \binom{d}{j} (-L)^j \tag{2}$$

The classical, and most popular, FARIMA model assumes that the error terms ε_t are normally distributed with zero mean and constant variance σ^2. However, since Internet traffic datasets suggest leptokurtic distributions, in addition to the classical model we also consider FARIMA models with Student's t innovations, i.e. the error terms are assumed to follow Student's t distribution with zero mean, constant variance σ^2 and v degrees of freedom where $v > 2$.

In order to construct a FARIMA model by fitting a traffic trace, a step-by-step procedure is followed (Katris & Daskalaki, 2014):

Step 1. The sample mean $\bar{x}$ of the traffic trace is subtracted from each observation in order to convert it to a zero-mean data series.

Step 2. The order (p, q) of the model is specified using the lowest Bayes Information Criterion (BIC), where $\text{BIC} = \ln(s^2) + n\ln(T)$, s^2 is the estimated variance for the residuals, n is the number of parameters to be estimated and T the number of observations. For our implementations, to be presented in the next Section, we restricted both the order of autoregressive as well as the order of the moving average components to be less than or equal to 5 $(0 \leqslant p \leqslant 5, 0 \leqslant q \leqslant 5)$.

Step 3. Having set the order of the model, the rest of the parameters (d, φ_i and θ_j) are estimated using the Geweke and Porter-Hudak (GPH) estimator for d and a Maximum Likelihood (ML) methodology for the others, under the assumption that the stationary, fractionally integrated series follows normal or alternatively Student's t distribution.

Geweke and Porter-Hudak (1983) proposed a non-parametric estimator for the fractional parameter d only, without specifying the other coefficients of the model. Alternatively, the recursive ML procedure suggested by Sowell (1992) can be used as a one-step procedure for all parameters (including d) or it can be applied solely for the estimation of φ_i and θ_j, if d is already known. Since the GPH estimator is computationally more efficient it is preferable to use it for d. However, in some cases it fails to give an estimate in $(0, 0.5)$, even though we may already know that the long-memory effect is present. In such cases one may choose to estimate d together with the remaining parameters, using the ML estimation methodology.

3.2. Artificial Neural Network Models

Since their introduction, neural networks have been successfully applied to many disciplines, including forecasting (Lippmann, 1987; Zhang, Patuwo, & Hu, 1998). They constitute recent approaches, compared to the classical time series models, and they can handle non-linear phenomena more successfully. ANN forecasting models for time series use sliding windows in the sense that a window with the k most recent values are used to predict the next one. There have been a number of different architectures for ANNs, however in this paper the MLP and RBF will be used. Fig. 1 displays a schematic view of a neural network.

In order to construct an ANN for time series one-step ahead prediction one needs to decide about the input variables, the number of hidden layers and number of nodes for each layer. Empirical research has shown that one hidden layer is sufficient in most cases; therefore we only have to define the number of input nodes and number of hidden nodes. For this task concepts from dynamical systems can be used (Frank et al., 2001). More specifically, the forecasting model is expressed as in Eq. (3):

$$x(t+1) = f(\mathbf{x}(t)) \tag{3}$$

where $\mathbf{x}(t) = (x(t),\ x(t-1\tau), \ldots, x(t-(n-1)\tau)$ is the vector of the lagged variables to be used as input for the forecast. It is assumed that the n-dimensional dynamic system from which the data come from is unknown, so the goal is to identify a simpler system from which the data could have come from. The embedding theorem (Mañé, 1981; Takens, 1981) claims that the space of time-lagged vectors with sufficiently large dimension (at least twice the dimension of the original space) can capture the original time series. This is the embedding dimension m and indicates the number of input nodes for the network. Furthermore, the time series data have to be independent in order to be transformed to m-dimensional vectors. We can achieve this by sampling past values with a sampling rate τ, where τ is the delay. We take $\tau > 1$ if the time series is considered oversampled. In practice, once we have decided the embedding dimension m, then $\hat{x}_{t+1} = f(x_t, x_{t-\tau}, \ldots, x_{t-(m-1)\tau})$, where τ is the time delay. Both parameters m and τ are important, since they define the length of the sliding window of the input variables, and need to be determined before proceeding with the predictions.

For the selection of the input nodes there are heuristics that can decide on an embedding dimension, such as the False Nearest Neighbors (FNN) method (Abarbanel & Kennel, 1993), which is used later in our study. For the resampling, on the other hand, there are a couple different methods that can help us decide about the time lag τ, but in this work the concept of Mutual Information has been used (Kantz & Schreiber, 2004). This procedure, which can be though as the analog of the correlation function but in a non-linear setting, uses the Mutual Information function $I_\varepsilon(t)$ and the scope is to achieve the lowest average mutual information between variables. The first local minimum of $I_\varepsilon(t)$ found is considered to be the resampling rate, however it is important to keep a balance between smaller τ and greater independence between variables. For example, if the minimum of $I_\varepsilon(t)$ has been found at lag 10, the series should be resampled every 10 periods, however, if the 1st lag leads to a very large decrease of $I_\varepsilon(t)$, not necessarily to a minimum, then it is preferable to keep the sampling rate of 1 and not to resample the data.

The MLP neural network used in this paper is a feed-forward ANN comprising of an input layer, one hidden layer and an output node. Each layer is fully connected to the next one and the activation function used in the hidden layer is the sigmoid:

$$S(t) = \frac{1}{1 + e^{-t}}.$$

Moreover, a linear function is used in the output layer in order to transform the previous inputs to final outputs. The training of the network has been done with the back-propagation technique, where the weights of the connections in the neural network are updated using the adaptive gradient descent optimization algorithm (Haykin, 1999).

Likewise, the RBF neural networks used are also feed-forward networks comprising of an input layer, one hidden layer with a non-linear RBF activation function and an output node. For this

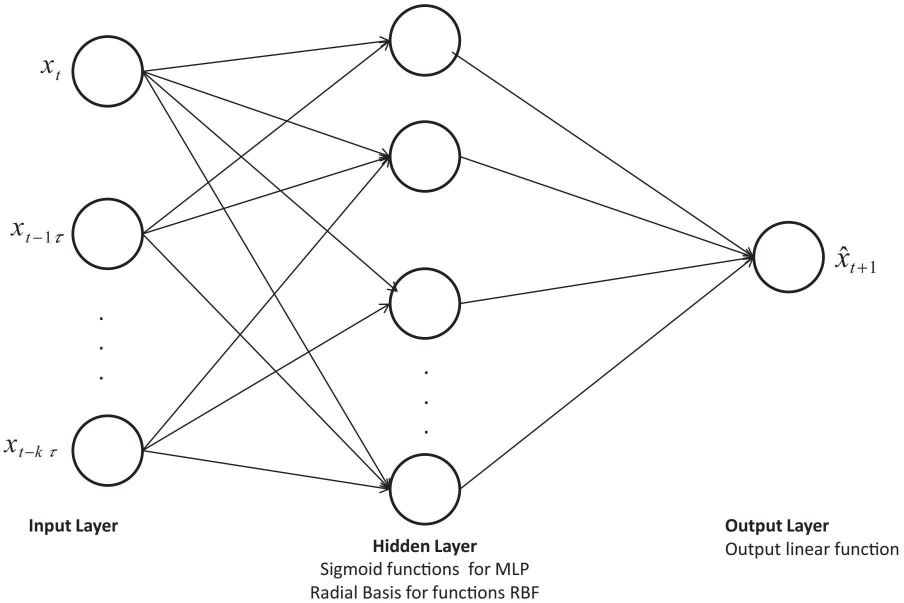


Fig. 1. The MLP neural network architecture.

work the Gaussian basis function in Eq. (4) has been used as activation function

$$\varphi(x, c, \sigma) = \exp\left(-\|x - c\|^2 / 2\sigma^2\right), \tag{4}$$

with center $c \in R^m$ being the mean and σ the width of the function. Finally, a linear transformation is performed going from the hidden layer to the output. The output of the RBF neural network is given as in Eq. (5)

$$\hat{y} = \sum_{i=1}^{N} w_i \varphi(x, c, \sigma) + w_0 \tag{5}$$

where w_0 is a bias term, w_i are the weights between hidden and output layers and N is the number of hidden neurons. During training, first the centroids are determined, one for every hidden node, and afterwards the corresponding widths. Last, the weights are determined using back-propagation (http://www.ra.cs.uni-tuebingen.de/SNNS/UserManual/UserManual.html). More details about training of RBF networks can be found in Schwenker, Kestler, and Palm (2001), while network traffic applications with RBF appear in Gowrishankar and Satyanarayana (2009), Szmit et al. (2013).

3.3. Model selection based on a non-linearity test

Both categories of forecasting models presented in Sections 3.1 and 3.2 are suitable for Internet traffic since FARIMA can model LRD and SRD, while ANNs can model more successfully non-linear dependencies. Unfortunately, as it became obvious from our experimental efforts, not all datasets carry non-linear structures. Therefore, a statistical test that can identify ahead of time such differences is undoubtedly useful for the selection of that model and can lead to better forecasts.

The WNN test (Lee et al., 1993; White, 1989) is one such test. For the time series itself it considers an augmented single hidden layer feed-forward ANN with p lags as input nodes and q hidden nodes, where the output y_t is determined by the input $\mathbf{x}_t$ according to the Eq. (6)

$$y_t = \mathbf{x}_t^T \boldsymbol{\beta} + \sum_{j=1}^{q} \delta_j G\left(\mathbf{x}_t^T \boldsymbol{\gamma}_j\right) + \varepsilon_t, \quad t = 1, \ldots, n \tag{6}$$

where $\mathbf{x}_t^T = (1, y_{t-1}, \ldots, y_{t-p})$, $\boldsymbol{\beta}$ is a conformable column vector of connection strengths from input layer to output layer, $\boldsymbol{\gamma}_j$ is a conformable column vector of connection strengths from input layer to the jth hidden unit, δ_j is a (scalar) connection strength from the jth hidden unit to output unit, with $j = 1, \ldots, q$ and G is an activation function (e.g. the logistic or radial function). The input units in $\mathbf{x}_t^T$ send signals to the intermediate hidden units, then each hidden unit produces an activation G that sends signals towards the output unit. The null hypothesis of the test is that the data carry a linear structure in the mean. Details and comparison of the WNN test with other similar tests can be found in Franses and Van Dijk (2000), Lee (2001).

For the Internet traffic forecasting problem, we suggest a procedure that follows two stages. At the *first stage* the WNN test is performed in order to detect existence of non-linearity in the data. If the test is significant we reject the linear hypothesis and conclude that there is a need for non-linear terms in the model. Otherwise, we conclude that the linear terms in the model are sufficient for the forecasting. Then based on this information at the *second stage* we select the appropriate model, which for the non-linear case would be a MLP neural network model and for the linear case would be a FARIMA model.

The implementation of such a procedure assumes that initially there will be sufficient data for testing non-linearity. Then the data can be used for training the MLP model in case of significance or for fitting the FARIMA in case of non-significance. Use of the suitable model is implemented from there on for the one-step ahead forecasts. The training or fitting of the corresponding models follow the procedures that were discussed in Sections 3.1 and 3.2. For our experimental work in Section 5, we consider both FARIMA with Normal and FARIMA with Student's t innovations and compare their performance. Such a procedure identifies the best individual forecasting model for each trace; therefore if the White test is powerful enough, the procedure may lead to a successful individual model.

3.4. FARIMA–ANN hybrid models

As an alternative to the previous approach we now present the construction of hybrid models that combine constructively

Please cite this article in press as: Katris, C., & Daskalaki, S. Comparing forecasting approaches for Internet traffic. *Expert Systems with Applications* (2015), http://dx.doi.org/10.1016/j.eswa.2015.06.029

FARIMA and ANNs. With hybridization one may benefit from both types of models by modeling at the same time long and short dependence but also non-linear structures, whenever they are present. The strategy here will be to combine a linear model with a neural network by first fitting a FARIMA model to the data and afterwards construct a neural network with the residuals.

It is known that Internet traffic data exhibit long-memory and FARIMA models are considered as appropriate, so in the first step a FARIMA model is fit to each dataset, following the procedure described in Section 3.1, with the aim to capture the linear part of the process generating the traffic. In the second step a neural network is built following the procedure described in Section 3.2, this time however using the time series of the residuals from the fitted FARIMA. At each step the forecasts of the two models are added together in a single forecast for one period ahead. Thus the forecast $\hat{y}_t$ at each epoch t is given as in Eq. (7):

$$\hat{y}_t = \hat{y}_t^L + \hat{y}_t^{NL} \tag{7}$$

where $\hat{y}_t^L$ and $\hat{y}_t^{NL}$ are the forecasted values from the FARIMA and ANN models, respectively.

For Internet traffic we need models that work well even for the cases where there exists an even weak non-linear relation compared to the linear one. In such cases, we believe that a hybrid model will be quite appropriate and can lead to improved forecasts.

4. Comparing forecasting models

Accuracy in forecasting is measured with a number of different metrics, with the most popular being the RMSE, MAE and MAPE, as shown in Eqs. (8)–(10).

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - f_i)^2} \tag{8}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|f_i - y_i| \tag{9}$$

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - f_i}{y_i}\right| \tag{10}$$

where n is the total number of forecasts, y_i the observed value at time i and f_i the respective forecasted value.

All three metrics measure the forecasting error when compared to the actual values and so report on the performance of each model. Smaller values would indicate better performance. The RMSE criterion is the most popular criterion for time series comparison. It depends on the scale of the variable of interest, thus is suitable for comparing different models across the same time series. Because of its quadratic nature large errors weigh higher and this fact makes RMSE more suitable when large errors are particularly undesirable. According to Chai and Draxler (2014), RMSE is sensitive to outliers and when outliers are not normally distributed RMSE could be biased. MAE on the other hand, depends also on the scale of the variable, but due to its linear nature all errors are weighted equally, and as a consequence is less sensitive to large errors. MAE is reported as preferred than RMSE, by Willmott and Matsuura (2005) as a more natural measure of average error. Finally, MAPE is another popular measure of accuracy because it is scale independent and also because it can be interpreted and understood better. Reported disadvantages of MAPE are associated with instabilities, when the original time series carries small values, and with asymmetrical penalties applied to positive errors compared to the negative ones.

However, when a number of forecasting models or approaches are considered for prediction on several datasets, the decision regarding the most appropriate model is not an easy task. In our case we are interested in adopting the best approach for Internet traffic and for this purpose we will use nine different datasets to test the performance of several different approaches. Therefore we set up a framework for comparison that involves first the ranking of the models according to the above measures and then calculation of the average position and its corresponding standard deviation.

Ranking of models: suppose there are N forecasting models which are applied to k datasets. For each dataset all models are ranked according to their RMSE and MAE values. Then we use their rank to calculate in Eq. (11) the average position (*AP*) of each model, which summarizes its performance over all datasets, and in Eq. (12) the standard deviation of the positions (*SDP*), to estimate the variability of the positions that a model has across the different datasets. Therefore, for each model i we calculate

$$AP_i = \frac{1}{k}\sum_{j=1}^{k}P_{ij} \tag{11}$$

where P_{ij} denotes the position of model i for dataset j, and

$$SDP_i = \sqrt{\frac{1}{k}\sum_{j=1}^{k}(P_{ij} - AP_i)^2} \tag{12}$$

It is obvious that one would prefer a model with the lowest possible *AP* and *SDP* at the same time. Since the problem has two objectives we introduce a linear utility function for the evaluation of models. For model i the utility function U is defined as in Eq. (13):

$$U_i = a_1 AP_i + a_2 SDP_i \tag{13}$$

where a_i denotes the weight of the corresponding objective on the utility function. In applications where the researchers have reasons to believe that the average position or the stability of a model is more important can give extra weight to the specific attribute, otherwise a_1 and a_2 will have the same value. For our experimental work in Section 5, we set $a_1 = a_2 = 1$.

Utility values can be assigned to models for each one of the performance measures used (i.e. RMSE, MAE and MAPE). So in order to assign a single evaluation metric for each model, one may think of an *average ranking value* which is calculated as in Eq. (14):

$$ARv_i = \frac{1}{3}\left(U_i^{\mathrm{RMSE}} + U_i^{\mathrm{MAE}} + U_i^{\mathrm{MAPE}}\right) \tag{14}$$

where a smaller ARv value for a model indicates a more accurate forecasting model.

5. Data analysis for Internet traffic forecasting

All previously presented forecasting approaches were applied to nine different datasets. The datasets were extracted out of real traffic traces, publicly available either from the Internet traffic archive (http://ita.ee.lbl.gov/) or from TU-Berlin (Fitzek & Reisslein, 2001). These nine datasets were used to evaluate and compare the traffic forecasting approaches presented earlier. We use video traces of high quality and TCP/IP packets for WAN traffic traces. More details about the data formatting can be found either in the TU-Berlin or the Internet traffic archive.

Table 1 gives an overview of the datasets used in this experimental study. For each dataset we give the name of trace, its source and traffic type. Since the datasets were created by us using the corresponding traces we also report the aggregation level we chose for each one, i.e. the unit measure for the load and the period of time used for the aggregation. As one may see the datasets are

Please cite this article in press as: Katris, C., & Daskalaki, S. Comparing forecasting approaches for Internet traffic. *Expert Systems with Applications* (2015), http://dx.doi.org/10.1016/j.eswa.2015.06.029

Table 1
Overview of datasets.

Trace	Source	Type	Unit	Aggregation time	Training	Test
Aug 89	Bellcore	LAN	Mbytes	10 s	264	50
Oct 89	Bellcore	LAN	Mbytes	Seconds	1260	500
LBL_PK4	Bellcore	WAN	Mbytes	10 s	300	60
Dusk till Dawn	TU Berlin	MPEG4	Kbytes	Frames	4000	1000
Die Hard III	TU Berlin	MPEG4	Kbytes	Frames	4000	1000
Jurassic Park	TU Berlin	MPEG4	Mbytes	Seconds	3000	600
Star Wars IV	TU Berlin	MPEG4	Mbytes	Seconds	3000	600
The Firm	TU Berlin	H.263	Kbytes	Frames	4000	1000
Mr. Bean	TU Berlin	H.263	Kbytes	Frames	4000	1000

Table 2
Descriptive statistics for all datasets.

Trace	Mean	St. Dev.	Skewness	Kurtosis	CV
Aug 89	1.3174	0.5657	1.0613	4.1318	0.4294
Oct 89	0.2990	0.1173	0.5133	3.5531	0.3925
LBL_PK4	0.3637	0.2139	1.5893	5.5847	0.5882
Dusk till Dawn	3.1838	2.056	1.4073	6.6582	0.6456
Die Hard III	3.1248	2.3684	1.4953	5.3769	0.7579
Jurassic Park	0.1002	0.0452	0.9637	4.2010	0.4507
Star Wars IV	0.0335	0.0128	1.1006	6.4343	0.3823
The Firm	1.614	1.2616	1.8510	6.8871	0.7817
Mr. Bean	2.2049	1.0725	2.3069	13.8533	0.4864

Table 4
Tests for LRD and non-linearity.

Trace	Hurst exponent	White test	(*p*-Value)	Suggested structure
Aug 89	0.6032	1.2628	(0.532)	Linear
Oct 89	0.9104	7.0662	(0.029)	Non-linear
LBL_PK4	0.8416	18.2919	(<0.01)	Non-linear
Dusk till Dawn	0.5772	177.2801	(<0.01)	Non-linear
Die Hard III	0.6979	231.4768	(<0.01)	Non-linear
Jurassic Park	0.9747	0.4978	(0.779)	Linear
Star Wars IV	0.8147	4.8190	(0.090)	Linear
The Firm	0.8253	3.1556	(0.206)	Linear
Mr. Bean	0.7888	53.447	(<0.01)	Non-linear

created from either Ethernet traffic (LAN and WAN traces) or Video traffic (MPEG4 and H.263 traces) and different aggregation levels were used, from Kbytes/frame and Mbytes/s to Mbytes/10 s. The last two columns refer to the volume of each dataset and specifically they give the split we used for the training and testing sets.

For each dataset initially there was an exploratory analysis and Table 2 displays the results, i.e. the mean, standard deviation, skewness, kurtosis and coefficient of variation. It can be seen that all datasets indicate right skewed and leptokurtic distributions, while the coefficient of variation varies from a low .38 to a high .78. Next the datasets were tested for normality, stationarity, randomness and autocorrelation and Table 3 displays the results from the corresponding tests.

Table 3 gives the values for the test statistics used for each test and the corresponding *p*-values in parenthesis. According to the results of the tests all datasets deviate significantly from normality. Such an observation suggests that the classical FARIMA models which assume normal distribution for the errors may be insufficient, therefore for our study we additionally considered models with Student-*t* innovations. Based on the runs tests, we conclude that for all traces data are non-random, therefore it is important to use models than can describe their structures, so that predicting future values of the time series can be more successful. Lastly, from the ADF and Ljung–Box tests we can infer that all datasets are stationary and auto-correlated. The existence of significant autocorrelation indicates further that models with a well designed dependence structure can capture this aspect of traffic.

Next, the datasets were examined for the properties LRD and non-linearity. For tracing long memory the Hurst exponent was

Table 3
Statistical tests for all datasets.

Trace	Normality Jarque–Bera	Randomness runs test	Stationarity ADF test*	Autocorrelation Ljung–Box**
Aug 89	63.645 (<0.01)	−7.6462 (<0.01)	−4.1061 Lag = 6, (<0.01)	101.593 (<0.01)
Oct 89	71.396 (<0.01)	−12.4569 (<0.01)	−4.9354 Lag = 10, (<0.01)	526.957 (<0.01)
LBL_PK4	185.755 (<0.01)	−7.0555 (<0.01)	−5.3965 Lag = 6, (<0.01)	81.923 (<0.01)
Dusk till Dawn	3550.683 (<0.01)	−31.3105 (<0.01)	−3.7648 Lag = 15,(0.02)	1612.094 (<0.01)
Die Hard III	2432.254 (<0.01)	−38.4265 (<0.01)	−4.364 Lag = 15, (<0.01)	1729.362 (<0.01)
Jurassic Park	644.692 (<0.01)	−41.5973 (<0.01)	−8.7937 Lag = 14, (<0.01)	2402.893 (<0.01)
Star Wars IV	2079.979 (<0.01)	−37.6166 (<0.01)	−8.2423 Lag = 14, (<0.01)	2074.100 (<0.01)
The Firm	4802.478 (<0.01)	−53.9868 (<0.01)	−6.6891 Lag = 15, (<0.01)	3179.949 (<0.01)
Mr. Bean	23180.250 (<0.01)	−41.0199 (<0.01)	−7.9354 Lag = 15, (<0.01)	1685.856 (<0.01)

* Alternative hypothesis for ADF test is stationarity.
** We are testing whether data are auto-correlated with lag = 1.

Table 5
Parameters of the forecasting Models.

Trace	FARIMA-N or *t*	RBF	MLP	FARIMA/RBF	FARIMA/MLP	Selected model
Aug 89	(1,*d*,1) *d* = 0.3961873	(4,1,1)	(4,20,1)	(1,*d*,1)/(7,5,1)	(1,*d*,1)/(7,1,1)	FARIMA
Oct 89	(1,*d*,1) *d* = 0.440625	(9,20,1)	(9,50,1)	(1,*d*,1)/(8,5,1)	(1,*d*,1)/(8,50,1)	MLP
LBL_PK4	(0,*d*,4) *d* = 0.3955	(7,50,1)	(7,50,1)	(0,*d*,4)/(7,5,1)	(0,*d*,4)/(7,1,1)	MLP
Dusk till Dawn	(4,*d*,2) *d* = 1e-08	(6,10,1)	(6,10,1)	(4,*d*,2)/(7,5,1)	(4,*d*,2)/(7,50,1)	MLP
Die Hard III	(5,*d*,2) *d* = 1e−08	(10,50,1)	(10,20,1)	(5,*d*,2)/(10,5,1)	(5,*d*,2)/(10,1,1)	MLP
Jurassic Park	(1,*d*,2) *d* = 0.2732444	(10,5,1)	(10,20,1)	(1,*d*,2)/(5,5,1)	(1,*d*,2)/(5,10,1)	FARIMA
Star Wars IV	(2,*d*,3) *d* = 0.1785253	(10,2,1)	(10,50,1)	(2,*d*,3)/(6,5,1)	(2,*d*,3)/(6,1,1)	FARIMA
The Firm	(0,*d*,1) *d* = 0.5	(6,10,1)	(6,10,1)	(0,*d*,1)/(5,5,1)	(0,*d*,1)/(5,20,1)	FARIMA
Mr. Bean	(2,*d*,2) *d* = 0.3777264	(10,20,1)	(10,10,1)	(2,*d*,2)/(5,5,1)	(2,*d*,2)/(5,50,1)	MLP

calculated using the R/S method, while for non-linearity the WNN test was performed. The results are displayed in Table 4. It can be observed that long memory is present in all datasets, with the weaker presence being in the datasets ‘*From Dusk till Dawn*’ and ‘*Aug 89*’. On the other hand, in six traces linearity was rejected (at the 0.05 significance level) and ANN will be the proposed modeling approach, while for the remaining traces linearity fails to be rejected and a FARIMA model is proposed.

Table 6
Performance Evaluation of Forecasting Models.

Trace		FARIMA-N	FARIMA-*t*	RBF	MLP	Hybrid FARIMA + RBF	Hybrid FARIMA + MLP
Aug 89	RMSE	0.4274	0.4244	0.4452	0.4460	0.4333	0.4286
	MAE	0.3211	0.3192	0.3358	0.3418	0.3213	0.3224
	MAPE	0.1854	0.1842	0.1912	0.1873	0.1844	0.1856
Oct 89	RMSE	0.1087	0.1063	0.1102	0.1077	0.1076	0.1057
	MAE	0.0882	0.0857	0.0890	0.0852	0.0873	0.0852
	MAPE	0.1781	0.1789	0.1789	0.1710	0.1784	0.1792
LBL_PK4	RMSE	0.2059	0.2221	0.2096	0.1894	0.2021	0.2051
	MAE	0.1468	0.1491	0.1521	0.1421	0.1458	0.1454
	MAPE	0.6086	0.5811	0.6694	0.6590	0.6393	0.5950
From Dusk till Dawn	RMSE	1.0259	1.1263	1.2689	1.0146	0.9518	0.8305
	MAE	0.7024	0.6409	0.8145	0.5899	0.6479	0.5165
	MAPE	0.2134	0.1922	0.2451	0.1639	0.2018	0.1638
Die Hard III	RMSE	0.7544	0.7356	0.7088	0.5135	0.6843	0.6498
	MAE	0.5290	0.4604	0.4795	0.3045	0.4958	0.4584
	MAPE	0.2547	0.2079	0.2203	0.1398	0.2517	0.2298
Jurassic Park	RMSE	0.0175	0.0179	0.0294	0.0427	0.0182	0.0177
	MAE	0.0123	0.0123	0.0235	0.0374	0.0133	0.0125
	MAPE	0.1990	0.1939	0.3536	0.7514	0.2260	0.2050
Star Wars IV	RMSE	0.0087	0.0090	0.0359	0.0123	0.0088	0.0088
	MAE	0.0053	0.0053	0.0342	0.0092	0.0053	0.0053
	MAPE	0.1671	0.1629	0.7350	0.2651	0.1763	0.1657
The Firm	RMSE	0.3961	0.4302	0.3962	0.3712	0.3902	0.3692
	MAE	0.1870	0.1865	0.2084	0.1751	0.1856	0.1743
	MAPE	0.2045	0.1983	0.2455	0.1895	0.2000	0.1829
Mr. Bean	RMSE	0.9156	0.9970	0.9346	0.8937	0.8960	0.8832
	MAE	0.4631	0.4345	0.4691	0.4248	0.4388	0.3979
	MAPE	0.2041	0.1769	0.2044	0.1721	0.1883	0.1622
RMSE evaluation	Avg. Position	3.44	4.22	5.22	3.33	2.94	1.83
	St. Dev.	1.74	1.86	0.67	2	0.73	0.87
	Value	**5.18**	**6.08**	**5.89**	**5.33**	**3.67**	**2.70**
MAE evaluation	Avg. Position	4	2.89	5.56	2.94	3.61	2.00
	St. Dev.	1.60	1.19	0.73	2.1	0.78	1.03
	Value	**5.60**	**4.08**	**6.29**	**5.04**	**4.39**	**3.03**
MAPE evaluation	Avg. Position	3.78	2.17	5.39	3.22	3.78	2.67
	St. Dev.	1.48	1.27	1.05	1.99	0.83	1.73
	Value	**5.26**	**3.44**	**6.44**	**5.21**	**4.61**	**4.40**
Model ranking		**5.35** (**7**)	**4.53** (**5**)	**6.20** (**8**)	**5.19** (**6**)	**4.22** (**4**)	**3.38** (**1**)

Please cite this article in press as: Katris, C., & Daskalaki, S. Comparing forecasting approaches for Internet traffic. *Expert Systems with Applications* (2015), http://dx.doi.org/10.1016/j.eswa.2015.06.029

Table 7
Performance Evaluation of the approach based on the White test.

		FARIMA-N or MLP	FARIMA-*t* or MLP
RMSE evaluation	Avg. Position	2.11	2.78
	St. Dev.	1.27	1.72
	Value	**3.38**	**4.49**
MAE evaluation	Avg. Position	2.06	1.83
	St. Dev.	1.21	0.97
	Value	**3.27**	**2.80**
MAPE evaluation	Avg. Position	2.89	2.33
	St. Dev.	1.69	1.66
	Value	**4.58**	**3.99**
Model ranking		**3.74** (**2**)	**3.76** (**3**)

Following the exploratory and preliminary tests, the models and forecasting approaches presented in Section 3 were applied to all datasets. We considered FARIMA models with Normal (FARIMA-N) and Student-*t* (FARIMA-*t*) innovations, RBF and MLP neural networks, hybrid models combining FARIMA-N with RBF or FARIMA-N with MLP and finally the procedure where the model is selected based on the non-linearity test. The construction of the models was performed using suitable routines from R. For FARIMA the package *rugarch* (Ghalanos, 2014) was used, where the maximum likelihood estimation requires the *nlminb* optimizer or the augmented Lagrange method for the nonlinear optimization procedures. For the MLP architecture the implementation was performed through the R package AMORE (Limas et al., 2014) and for the RBF architecture using the R package RSNNS (Bergmeir and Benitez, 2012). Finally, the WNN test, was implemented using the R package tseries (Trapletti & Hornik, 2013).

For the construction of the MLPs the embedding dimension was indicated using the FNN algorithm, while we experimented with 1, 2, 10, 20, or 50 nodes in the hidden layer and the final choice was based on a minimum RMSE during training, calculated for the 100 most recent observations of the training sample. The training was performed using back-propagation with the adaptive gradient descent algorithm and for 500 epochs of training. Lastly, the activation function was chosen to be sigmoid for the hidden layer and linear for the output. Similarly, for the RBF architecture, we decided to select five nodes for the hidden layer. After experimenting with 1, 5, 10, 20 and 50 nodes, with the selection of 5 nodes, there were no convergence problems in any trace. The training was performed using the "*RadialBasisLearning*" function from Stuttgart Neural Network Simulator (http://www.ra.cs.uni-tuebingen.de/SNNS/UserManual/UserManual.html). The activation function was Gaussian RBF for the hidden layer and linear for the output. For both architectures, the output layer consists of a single node, representing the one-period ahead prediction. Lastly, for the ANN models no resampling was suggested from the Mutual Information criterion, so τ was set be 1 for all traces.

The parameters of the models that resulted after applying the fitting process on the training set of each dataset are displayed in Table 5. Specifically, for each FARIMA model the orders *p* and *q* are shown, as well as the value of the fractional difference parameter *d*. Similarly, for the ANN models the number of nodes in each layer is displayed.

The selected models were used for one-step ahead forecasting and their performance according to the RMSE, MAE and MAPE criteria is displayed in Table 6. It must be noted that six separate approaches are displayed and for that one where the model is selected according to the White NN test the values each time follow those of the corresponding selected model. The performance of all approaches was then evaluated using the framework presented in Section 4. More specifically for each approach, its average position among the eight different forecasting approaches and the stability (standard deviation) of position are calculated as well as their average ranking values. Their ranking position is also reported in parentheses and the results are displayed across both Tables 6 and 7. Based on their average ranking values the individual models were ranked last, holding the positions 5–8, while the hybrid FARIMA–MLP model was ranked first and the hybrid FARIMA–RBF fourth. The 2nd and 3rd positions appear at Table 7 for the approach that selects a forecasting model based on the linearity test. Needless to say, the performance of the three best approaches is very close. The evaluation of the forecasting approaches is displayed also graphically in Fig. 2 where the values for $1/U_i^{\mathrm{RMSE}}$, $1/U_i^{\mathrm{MAE}}$, $1/U_i^{\mathrm{MAPE}}$ and $1/ARv_i$ respectively are displayed. This means that higher values indicate better performances.

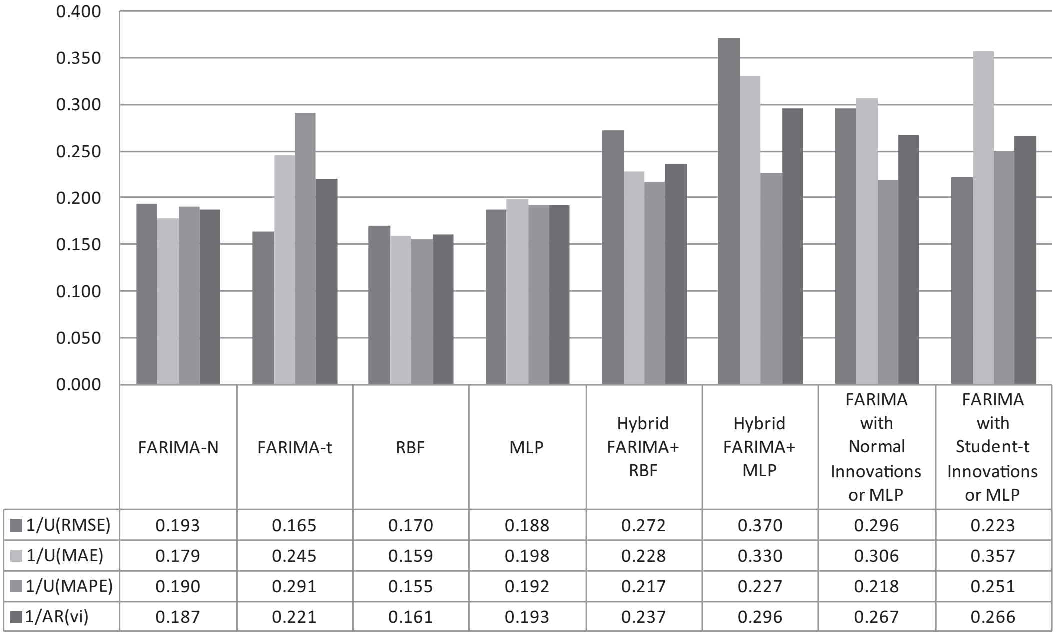
From Fig. 2, one may conclude that the FARIMA models with normal innovations and MLP ANNs display similar performance, giving 0.187 and 0.193 as the reciprocal of their ARv_i. It is also worth noticing that the FARIMA model with student-*t* innovations performs well according to MAPE and MAE, giving 0.291 and 0.245, respectively, but comparatively poor (0.165) according to RMSE. As mentioned in Section 4, poor performance in RMSE suggests the presence of some extreme deviations between the actual and forecasted values. The approach that uses the White test for selecting a model leads to an improved performance over all individual models, giving 0.267 and 0.266 for the reciprocal of their ARv_i, proving that neither FARIMA nor ANNs can always be the best choice for Internet traffic data series. Lastly, from the hybrid models the FARIMA–MLP outperformed all individual models and other approaches, while the FARIMA–RBF combination performs better than the individual models but worse than both cases based on the White test.

Following the analysis just presented, a more detailed one evaluated the models separately for the different types of traces, i.e. LAN and WAN (Ethernet traces), MPEG4 video and H.263 video. Table 8 presents the ranking values and ranking position (in parenthesis) of each model within each category. We observe that the approaches which used either the White test or hybrid models performed better than the individual models for all categories. The hybrid FARIMA + MLP is ranked first for the H.263 traces, and the two approaches with the White test are ranked first for the MPEG4 and Ethernet traces, respectively. Comparing the two FARIMA choices, one may comment that FARIMA-*t* gives better forecasts for the video traces and worse for the Ethernet traces. Similar comparison for the hybrid choices brings the RBF models

Table 8
Ranking of models within each category.

Category	FARIMA-N	FARIMA-*t*	RBF	MLP	Hybrid FARIMA + RBF	Hybrid FARIMA + MLP	FARIMA-N or MLP	FARIMA-*t* or MLP
Ethernet	4.54 (5)	4.94 (6)	6.17 (8)	5.92 (7)	3.97 (3)	4.44 (4)	3.62 (2)	3.28 (1)
MPEG4	5.71 (6)	3.71 (4)	6.28 (8)	5.9 (7)	4.46 (5)	3.09 (2)	3.04 (1)	3.52 (3)
H.263	4.67 (5)	4.40 (4)	5.67 (8)	2.00 (2)	3.74 (3)	1.00 (1)	5.22 (7)	4.82 (6)
Overall	**5.35** (**7**)	**4.53** (**5**)	**6.20** (**8**)	**5.19** (**6**)	**4.22** (**4**)	**3.38** (**1**)	**3.74** (**2**)	**3.76** (**3**)

Please cite this article in press as: Katris, C., & Daskalaki, S. Comparing forecasting approaches for Internet traffic. *Expert Systems with Applications* (2015), http://dx.doi.org/10.1016/j.eswa.2015.06.029



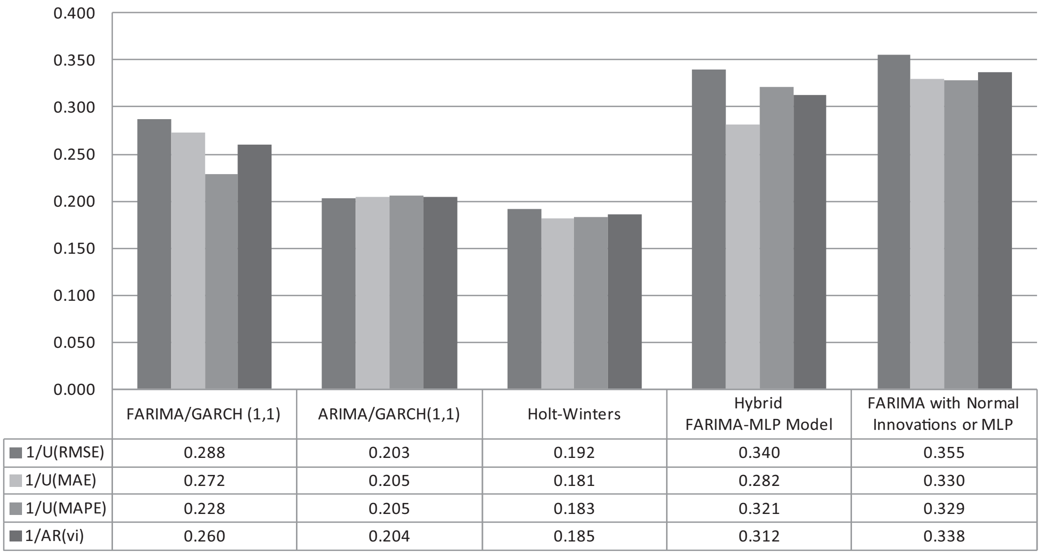
	FARIMA-N	FARIMA-t	RBF	MLP	Hybrid FARIMA+ RBF	Hybrid FARIMA+ MLP	FARIMA with Normal Innovations or MLP	FARIMA with Student-t Innovations or MLP
■ 1/U(RMSE)	0.193	0.165	0.170	0.188	0.272	0.370	0.296	0.223
■ 1/U(MAE)	0.179	0.245	0.159	0.198	0.228	0.330	0.306	0.357
■ 1/U(MAPE)	0.190	0.291	0.155	0.192	0.217	0.227	0.218	0.251
■ 1/AR(vi)	0.187	0.221	0.161	0.193	0.237	0.296	0.267	0.266

Fig. 2. Evaluation of forecasting approaches.

Table 9
Performance Evaluation of Forecasting Approaches.

Trace		FARIMA/GARCH	ARIMA/GARCH	Holt-Winters	Hybrid FARIMA + MLP	FARIMA-N or MLP
Aug 89	RMSE	0.4275	0.4219	0.4309	0.4286	0.4274
	MAE	0.3208	0.3159	0.3161	0.3224	0.3211
	MAPE	0.1834	0.1827	0.1923	0.1856	0.1854
Oct 89	RMSE	0.1062	0.1169	0.1120	0.1057	0.1077
	MAE	0.0855	0.0952	0.0899	0.0852	0.0852
	MAPE	0.1762	0.1858	0.1895	0.1792	0.1710
LBL_PK4	RMSE	0.2089	0.2052	0.4675	0.2051	0.1894
	MAE	0.1473	0.1528	0.2970	0.1454	0.1421
	MAPE	0.5974	0.6992	1.6026	0.5950	0.6590
From Dusk till Dawn	RMSE	0.9838	1.0292	1.4097	0.8305	1.0146
	MAE	0.6339	0.7147	0.9519	0.5165	0.5899
	MAPE	0.1853	0.2178	0.2883	0.1638	0.1639
Die Hard III	RMSE	0.7223	0.7592	0.9457	0.6498	0.5135
	MAE	0.4956	0.5413	0.6138	0.4584	0.3045
	MAPE	0.2317	0.2618	0.2710	0.2298	0.1398
Jurassic Park	RMSE	0.0176	0.0178	0.0257	0.0177	0.0175
	MAE	0.0124	0.0126	0.0188	0.0125	0.0123
	MAPE	0.2026	0.2091	0.2999	0.2050	0.1990
Star Wars IV	RMSE	0.0089	0.0089	0.0091	0.0088	0.0087
	MAE	0.0053	0.0053	0.0054	0.0053	0.0053
	MAPE	0.2091	0.1646	0.1677	0.1657	0.1671
The Firm	RMSE	0.3924	0.6448	0.4072	0.3692	0.3961
	MAE	0.1887	0.5413	0.1835	0.1743	0.1870
	MAPE	0.2079	0.8427	0.1859	0.1829	0.2045
Mr. Bean	RMSE	0.9154	0.9164	0.9485	0.8832	0.8937
	MAE	0.4631	0.4622	0.4694	0.3979	0.4248
	MAPE	0.2041	0.2029	0.1998	0.1622	0.1721
RMSE evaluation	Avg. Position	2.72	3.72	4.78	1.89	1.89
	St. Dev.	0.75	1.20	0.44	1.05	0.93
	Value	**3.48**	**4.92**	**5.22**	**2.94**	**2.82**
MAE evaluation	Avg. Position	2.94	3.61	4.22	2.22	2.00
	St. Dev.	0.73	1.27	1.30	1.33	1.03
	Value	**3.67**	**4.88**	**5.52**	**3.55**	**3.03**
MAPE evaluation	Avg. Position	3.11	3.44	4.33	2.00	2.11
	St. Dev.	1.27	1.42	1.12	1.12	0.93
	Value	**4.38**	**4.86**	**5.45**	**3.12**	**3.04**
Model ranking		**3.84** (**3**)	**4.89** (**4**)	**5.40** (**5**)	**3.20** (**2**)	**2.96** (**1**)

Please cite this article in press as: Katris, C., & Daskalaki, S. Comparing forecasting approaches for Internet traffic. *Expert Systems with Applications* (2015), http://dx.doi.org/10.1016/j.eswa.2015.06.029



	FARIMA/GARCH (1,1)	ARIMA/GARCH(1,1)	Holt-Winters	Hybrid FARIMA-MLP Model	FARIMA with Normal Innovations or MLP
■ 1/U(RMSE)	0.288	0.203	0.192	0.340	0.355
■ 1/U(MAE)	0.272	0.205	0.181	0.282	0.330
■ 1/U(MAPE)	0.228	0.205	0.183	0.321	0.329
■ 1/AR(vi)	0.260	0.204	0.185	0.312	0.338

Fig. 3. Evaluation of forecasting approaches.

Table 10
Ranking of models within each category.

Category	FARIMA/GARCH	ARIMA/GARCH	Holt–Winters	Hybrid FARIMA–MLP model	FARIMA–N or MLP
Ethernet	3.00 (1)	5.05 (4)	5.15 (5)	4.26 (3)	3.42 (2)
MPEG4	3.58 (3)	4.42 (4)	5.08 (5)	2.98 (2)	2.53 (1)
H.263	4.14 (3)	5.28 (5)	4.68 (4)	1.00 (1)	3.21 (2)
Overall	**3.84** (**3**)	**4.89** (**4**)	**5.40** (**5**)	**3.20** (**2**)	**2.96** (**1**)

to lower positions than MLP, which in the case of H.263 video traces is positioned second.

Considering the overall comparison, we conclude that the hybrid FARIMA + MLP and the FARIMA-N or MLP based on the White test are better strategies for implementing to forecast Internet traffic. As a next step in our experimentation we compare the two selected approaches with other models which have been used previously for Internet traffic prediction. We consider the ARIMA/GARCH process, used by (Zhou, 2006), and the classical Holt–Winters, used extensively for Internet traffic prediction such as in Cortez et al. (2012) . In addition we considered the FARIMA/GARCH model, which carry the desired advantages of a FARIMA and partially the advantages of ANN. The order of the GARCH model was taken to be (1,1), since at this order it can capture second order non-linearity. Table 9 gives the average position and the standard deviation of position for five different approaches, the two best ones from the previous comparison and the three new ones. In addition Fig. 3 gives the ranking of the models by plotting $1/U_i^{\mathrm{RMSE}}$, $1/U_i^{\mathrm{MAE}}$, $1/U_i^{\mathrm{MAPE}}$ and $1/ARv_i$.

From Table 9 and Fig. 3 we can conclude that the approaches with the White test and the hybrid FARIMA + MLP outperformed the other methods, while the FARIMA/GARCH performed better than ARIMA/GARCH and Holt–Winters. This can be explained since the hybrid FARIMA + MLP is more general than FARIMA/GARCH and can take into account non-linearities of higher order. ARIMA/GARCH, on the other hand, is more limited since it lacks long memory dependencies and higher order non-linearities. It is worth noticing that all criteria lead to the same ranking of the models.

Similarly to the results of Table 8 we performed a more detailed comparison of the models for the different categories of Internet traffic. Table 10 presents the ranking value and ranking position (in parenthesis) of each model within each category. We observe that the hybrid FARIMA + MLP, the White test approach and the FARIMA/GARCH hold the first position for the three different categories. Moreover, FARIMA/GARCH was ranked better than ARIMA/GARCH and Holt-Winters for all categories.

The experimentation presented in this paper explored the forecasting capabilities of certain approaches and indicated that the hybrid FARIMA + MLP and the approach with the White test perform better than the others. We now set the question whether such approaches can be implemented for online predictions. For such implementation one should consider that testing for non-linearity and the fitting or training procedures for any of the approaches should be done offline and then the use of the models for predicting traffic can be online. Keeping that in mind we measured the response times of used procedures. According to our measurements predictions using a plain MLP always required less than 0.001 s and can be implemented online even for frames that require 1/24 s. For the predictions using a FARIMA model, however, it was found that online predictions can be possible only for time periods greater than 0.3 s.

6. Summary and conclusions

To sum up, in this work we presented a set of forecasting approaches to be used for Internet traffic and a framework to compare them. The main idea was to take advantage of the statistical properties of Internet traffic and its possible non-linear structure and considered models that perform well under such conditions. We considered FARIMA models, MLP and RBF neural network architectures as individual models. Then, for further prediction improvement we proposed two alternative approaches which can combine the advantages of both FARIMA and neural network individual models. The first uses the White NN test to choose each time between models while the other embodies the two models into a hybrid FARIMA–ANN scheme. A framework for evaluation is also proposed since the criteria of RMSE and MAE were not sufficient by themselves for a fair comparison of the models. Finally, the experimental analysis presented was performed using publically available traces of Ethernet and video traffic prediction.

The results from the experimental work indicate that combining FARIMA models with neural network models is the best

Please cite this article in press as: Katris, C., & Daskalaki, S. Comparing forecasting approaches for Internet traffic. *Expert Systems with Applications* (2015), http://dx.doi.org/10.1016/j.eswa.2015.06.029

strategy when it comes to forecasting Internet traffic. Apparently, the key feature of this type of traffic is that non-linearity may or may not be present and the proposed hybridization works well since it fits first a FARIMA model and leaves the neural network for the residuals. However, choosing between a FARIMA and a neural network model each time based on a test for non-linearity is an equally successful approach and challenges the previous one quite well.

Both proposed procedures outperformed other approaches which have been used for Internet traffic, such as Holt–Winters and ARIMA/GARCH, or approaches which theoretically account for both long-memory and non-linearity, such as the FARIMA/GARCH model, when compared using a number of performance measures and a number of several traces.

Based on our results, we could outline certain directions for future research. The models have to be tested for more Internet traces datasets in order to validate the success of the hybrid FARIMA–MLP and the model selection approach according to White test. In our analysis we compared the models generally, for different types of traffic and in a future research it would interesting to study the suitability of approaches for a specific kind of traffic (e.g. MPEG 4 video traffic) and for different level of time aggregation would be interested. Another direction for continuing our research could be to consider different models for selection according to WNN test. Finally, a topic of future study is the improvement of hybrid FARIMA–MLP models using different hybridization strategies, such as unequal weight specifications for the linear and non-linear part of the models.

7. Uncited references

Zeng and Qiao (2011), Zhou et al. (2005).

References

Abarbanel, H., & Kennel, M. (1993). Local false nearest neighbors and dynamical dimensions from observed chaotic data. *Physical Review E, 47*, 3057–3068.

Aladag, C. H., Egrioglou, E., & Cadilar, C. (2012). Improvement in forecasting accuracy using the model of ARFIMA and feed forward neural network. *American Journal of Intelligent Systems, 2*, 12–17.

Alarcon-Aquino, V., & Barria, J. (2006). Multiresolution FIR neural-network-based learning algorithm applied to network traffic prediction. *IEEE Transactions on Systems, Man and Cybernetics – Part C, 36*, 208–220.

Balkin, S. D., & Ord, J. K. (2000). Automatic neural network modeling for univariate time series. *International Journal of Forecasting, 16*, 509–515.

Beran, J., Sherman, R., Taqqu, M. S., & Willinger, W. (1995). Variable bit-rate video traffic and long range dependence. *IEEE Transactions on Communications, 43*, 1566–1579.

Bergmeir, C., & Benitez, J. M. (2012). Neural networks in r using the Stuttgart neural network simulator: RSNNS. *Journal of Statistical Software, 46*(7), 1–26. URL <http://www.jstatsoft.org/v46/i07/>.

Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? *Geoscientific Model Development Discussions, 7*, 1525–1534.

Chiruvolu, G., Sankar, R. (1997). An approach towards resource management and transportation of VBR video. In *Proceedings of ICC97*.

Corradi, M., Garroppo, R.G., Giordano, S. & Pagano, M. (2001). Analysis of f-ARIMA processes in the modeling of broadband traffic. *ICC'01* (Vol. 3, pp. 964–968).

Cortez, P., Rio, M., Rocha, M., & Sousa, P. (2012). Multi-scale internet traffic forecasting using neural networks and time series methods. *Expert Systems, 29*(2), 143–155.

Dethe, C. G., & Wakde, D. G. (2004). On the prediction of packet process in network traffic using FARIMA time-series model. *Journal of the Indian Institute of Science, 84*, 31–39.

Dorffner, G. (1996). Neural networks for time series processing. *Neural Network World, 4*, 447–468.

Dymora, P., Mazurek, M., & Strzalka, D. (2013). Computer network traffic analysis with the use of statistical self-similarity factor. *Annales UMCS Informatica AI XIII, 2*, 69–81.

Edwards, T., Tansley, D. S. W., Davey, N., & Frank, R. J. (1997). Traffic trends analysis using neural networks. *Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications, 3*, 157–164.

Fitzek, F. H. P., & Reisslein, M. (2001). MPEG-4 and H.263 video traces for network performance evaluation. *IEEE Network, 15*(6), 40–54.

Frank, R. J., Davey, N., & Hunt, S. P. (2001). Time series predictions and neural networks. *Journal of Intelligent and Robotic Systems, 31*(1–3), 91–103.

Franses, P. H., & Van Dijk, D. (2000). *Non-linear time series models in empirical finance*. Cambridge University Press.

Geweke, G., & Porter-Hudak, S. (1983). The estimation and application of long memory time series models. *Journal of Time Series Analysis, 4*(4), 221–238.

Ghalanos, A. (2014). Rugarch: Univariate GARCH models. R packageversion 1.3-3.

Gowrishankar, S., & Satyanarayana, P. S. (2009). A time series modeling and prediction of wireless network traffic. *International Journal of Interactive Mobile Technologies, 3*(1), 53–62.

Granger, C. W. J., & Joyeux, R. (1980). An introduction to long-memory time series models and fractional differencing. *Journal of Time Series Analysis, 1*, 15–30.

Haykin, S. (1999). *Neural networks – a comprehensive foundation* (2nd ed.). 0-13-273350-1. New Jersey: Prentice Hall.

Hosking, J. R. M. (1981). Fractional differencing. *Biometrika, 68*, 165–176.

Hurst, H. E. (1951). Long-term storage capacity of reservoirs. *Transactions of the American Society of Civil Engineers, 116*, 770–808.

Jiang, J., & Papavassiliou, S. (2004). Detecting network attacks in the internet via statistical network traffic normality prediction. *Journal of Network and Systems Management, 12*, 51–72.

Kantz, H., & Schreiber, T. (2004). *Nonlinear time series analysis* (2nd ed.). Cambridge University Press.

Katris, C., & Daskalaki, S. (2014). Prediction of internet traffic using time series and neural networks. In *Proceedings of international work-conference on time series analysis (ITISE 2014)* (Vol. 1, pp. 594–605).

Lee, T. H. (2001). Neural network test and nonparametric kernel test for neglected nonlinearity in regression models. *Studies in Nonlinear Dynamics & Econometrics, 4*(4), 1–15.

Lee, T. H., White, H., & Granger, C. W. J. (1993). Testing for neglected nonlinearity in time series models. *Journal of Econometrics, 56*, 269–290.

Leland, W. E., Taqqu, M. S., Willinger, W., & Wilson, D. (1994). On the self-similar nature of ethernet traffic (extended version). *IEEE/ACM Transactions on Networking, 2*, 1–15.

Limas, M. C., Mere, J. B. O., Marcos, A. G., de PisonAscacibar, F. J. M., Espinoza, A. V. P., Elias, F. A., & Ramos, J. M. P. (2014). AMORE: A MORE flexible neural network package. R package version 0.2-15. <http://CRAN.R-project.org/package=AMORE>.

Lippmann, R. P. (1987). An introduction to computing with neural nets. *IEEE ASSP Magazine, 4*, 4–22.

Ma, L., & Xu, X. (2007). RBF network-based chaotic time series prediction and its application in foreign exchange market. In *Proceedings of the international conference on intelligent systems and knowledge engineering (ISKE 2007)*.

Mandelbrot, B. (1972). Statistical methodology for non-periodic cycles: From the covariance to R/S analysis. *Annals of Economic and Social Measurement, 1*, 257–288.

Mandelbrot, B. B., & Van Ness, J. W. (1968). Fractional Brownian motions, fractional noises and applications. *SIAM Review, 10*, 422–437.

Mañé, R. (1981). On the dimension of the compact invariant sets of certain nonlinear maps. In D. A. Rand & L.-S. Young (Eds.), *Dynamical systems and turbulence. Lecture notes in mathematics* (Vol. 898, pp. 230–242). Springer-Verlag.

Moussas, V. C., Daglis, M., & Kolega, E. (2005). Network traffic modeling and prediction using multiplicative seasonal ARIMA models. In D. T. Tsahalis (Ed.). *Proceedings of the 1stEpsMsO "international conference on experiments/process/system modelling/simulation/optimization"* (Vol. II, pp. 698–704). Patras: Patras University Press. Athens, 6–9 July, 2005, ISBN: 9605300869.

Patterson, D. W., Chan, K. H., Tan, C. M. (1993). Time series forecasting with neural nets: A comparative study. In *Proceedings of the international conference on neural network applications to signal processing. NNASP 1993* (pp. 269–274). Singapore.

Peters, E. E. (1994). *Fractal market analysis: Applying chaos theory to investment and economics*. Brisbane: John Wiley and Sons Inc.

Rutka, G. (2009). Some aspects of traffic analysis used for internet traffic prediction. *Electronics and Electrical Engineering Kaunas: Technologija, 5*(93), 7–10.

Schwenker, F., Kestler, H. A., & Palm, G. (2001). Three learning phases for radial-basis-function networks. *Neural Networks, 14*(4), 439–458.

Shu, Y., Jin, Z., Zhang, L., Wang, L., Oliver, W., & Yang W. (1999). Traffic prediction using FARIMA models. In: *IEEE international conference on communications* (Vol. 2, pp. 891–895).

Sowell, F. (1992). Maximum likelihood estimation of stationary univariate fractionally integrated time series models. *Journal of Econometrics, 53*, 165–188.

Szmit, M., Szmit, A., & Kuzia, M. (2013). Usage of RBF networks in prediction of network traffic. In *Position papers of the 2013 federated conference on computer science and information systems* (pp. 63–66).

Takahashi, Y., Aida, H., & Saito, T. (2000). ARIMA model's superiority over f-ARIMA model. In *International conference on communication technology proceedings, WCC – ICCT 2000* (Vol. 1, pp. 66–69).

Takens, F. (1981). Detecting strange attractors in turbulence. In D. A. Rand & L.-S. Young (Eds.), *Dynamical systems and turbulence. Lecture notes in mathematics* (Vol. 898, pp. 366–381). Springer-Verlag.

Trapletti, A., & Hornik, K. (2013). tseries: Time series analysis and computational finance. R package version 0.10-32.

Wang, C., Zhang, X., Yan, H., & Zheng, L. (2008). An internet traffic forecasting model adopting radial based on function neural network optimized by genetic algorithm, In *Proceedings of IEEE workshop on knowledge discovery and data mining (WKDD08)* (pp. 367–370). Adelaide, Australia.

White, H. (1989). An additional hidden unit tests for neglected nonlinearity in multilayer feedforward networks. *Proceedings of the international joint*

Please cite this article in press as: Katris, C., & Daskalaki, S. Comparing forecasting approaches for Internet traffic. *Expert Systems with Applications* (2015), http://dx.doi.org/10.1016/j.eswa.2015.06.029

conference on neural networks (Vol. II, pp. 451–455). New York, NY, Washington DC: IEEE Press.

Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research, 30*(1), 79.

Won, Y., & Ahn, S. (2005). GOP ARIMA: Modeling the nonstationarity of VBR processes. *Multimedia Systems, 10*(5), 359–378.

Zeng, J., & Qiao, W. (2011). Short-term solar power prediction using an RBF neural network. In *2011 IEEE power and energy society general meeting* (pp. 1–8).

Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing, 50*, 159–175.

Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting, 14*(1), 35–62.

Zhani, M. F., Elbiase, H., & Farouk, K. (2009). Analysis and prediction of real network traffic. *Journal of Networks, 4*(9), 855–865.

Zhou, B., He, D., Sun, Z., & Ng, W. H. (2005). Network traffic modeling and prediction with ARIMA/GARCH. *Proceedings of HET-NETs conference*, 1–10.