Chaudhry *J Wireless Com Network* (2021) 2021:200
https://doi.org/10.1186/s13638-021-02076-1

EURASIP Journal on Wireless
Communications and Networking

RESEARCH **Open Access**

Using machine learning to find the hidden relationship between RTT and TCP throughput in WiFi

Aizaz U. Chaudhry*

*Correspondence:
auhchaud@sce.carleton.ca
Department of Systems
and Computer Engineering,
Carleton University,
Ottawa K1S 5B6, Canada

Abstract

Is it possible to find hidden relationships among variables in WiFi network using *machine learning* (ML)? Can we use ML to find a variable that significantly affects the TCP throughput in WiFi? In this work, we employ a publicly available WiFi dataset to investigate these questions. We use ML techniques, including *principal component analysis* (PCA), *linear regression* (LR), and *random forest* (RF), to study the effect of link speed, received signal strength, *round-trip time* (RTT), and number of available access points on TCP throughput in WiFi. More specifically, we are interested in employing ML to find the variable that most accurately predicts and thereby most significantly affects the throughput. Simple correlation analysis indicates that a combination of multiple variables is more likely to act as a reasonable predictor of the throughput, whereas a single variable, such as RTT, alone is not likely to predict the throughput with reasonable accuracy. From PCA, the *first principal component* (PC1) is seen as highly correlated to RTT. During predictive analysis, it is observed that the LR model is unable to find any hidden relationship between throughput and other variables. However, the RF model discovers that RTT explains the variation in throughput more closely and as such it predicts the throughput more accurately compared to other variables. PC1 captures nearly all of the variation in throughput with the RF model and predicts throughput with very high accuracy, which indirectly confirms RTT as the variable that most significantly affects the TCP throughput in WiFi. Consequently, we discover a very close relationship between RTT and TCP throughput using appropriate ML techniques, and these results can be helpful in developing a better understanding of the relationship between latency and throughput for designing future low-latency networks.

Keywords: Machine learning, Random forest, Round-trip time, TCP throughput, WiFi

1 Introduction

WiFi is widely deployed and will continue to play a significant role in the presence of next generation cellular networks like 5G. One of the pillars of 5G is the support of *ultra-low latency* (ULL) applications such as augmented reality, virtual reality, and telemedicine. Latency or delay in the network is the primary concern for these applications, and it must be minimized to enable such applications. However, if WiFi is included in an ULL link, there could be large and variable delays due to the inherent WiFi protocols. The round-trip delay in TCP—i.e., the delay encountered by the packet to reach the destination plus the delay experienced by the acknowledgement of that packet to arrive at the source—includes the WiFi-induced delay as well the delay incurred over the Internet, and we refer to it as the *round-trip time* (RTT). Considering these upcoming ULL applications, the effect of RTT on TCP throughput in WiFi networks is of significant interest. Consequently, RTT is one of the variables whose relationship with TCP throughput in WiFi is being investigated in this paper, and a better understanding of the relationship between RTT and TCP throughput can be beneficial in advancing the knowledge of the relationship between latency and throughput for designing better low-latency networks.

In this work, we aim to use *machine learning* (ML) to find hidden relationships among variables within WiFi networks. We study the effect of various variables in the WiFi network, such as link speed, received signal strength, RTT, and number of available access points, on TCP throughput in WiFi. The WiFi dataset that we use for this purpose is a subset of the publicly available *Cell vs WiFi* dataset [1]. Our goal is to employ ML techniques, such as *linear regression* (LR), *random forest* (RF), and *principal component analysis* (PCA), to find the variable that most significantly affects the TCP throughput in WiFi.

We conduct three different types of analyses on the WiFi dataset including correlation analysis, principal component analysis, and predictive analysis. In correlation analysis, we simply check for visible correlation between different variables and the throughput via pairwise scatter plots and correlations. We also examine the correlations between the original variables and *principal components* (PCs) as well as proportions of variance for PCs. Before performing predictive analysis, we divide the available data into a training set and a test set. The performance of a variable or a combination of variables or a PC used to build the model using LR or RF for predicting the throughput is compared in terms of *percentage of variance* (PoV) between the actual training set and fitted values, and *root mean square error* (RMSE) between the actual test set and predicted values.

Visual observations of the results of correlation analysis indicate that link speed or received signal strength alone or in combination with other variables are likely to act as a reasonable predictor of TCP throughput in WiFi, whereas other individual variables, such as RTT or number of available access points, alone are not likely to reasonably predict throughput. The principal component analysis reveals that the *first principal component* (PC1) is highly correlated to RTT. During predictive analysis, a PoV of 24.49% is captured when all four variables are used by the LR model as compared to a PoV of 4.27% when only RTT is used. When used in combination with the RF model to produce fitted and predicted values for TCP throughput, RTT achieves a PoV of 87.63% and a RMSE of 1.33. This result outperforms all single variables as well as their combinations. When PC1 is used in generating the RF model, a very high PoV of 99.61% and a very low RMSE of 0.23 are observed. This leads us to conclude that RTT significantly impacts the TCP throughput in WiFi, and we can state the main contribution of this work as follows. *To the best of our knowledge, our work is the first effort of its kind to apply ML on a publicly available WiFi dataset to discover RTT as the variable amongst other variables that most accurately predicts and thereby most significantly affects TCP throughput in WiFi.*

It also reveals that employing RF in combination with PCA is highly beneficial in finding hidden relationship between variables.

Preliminary work in this regard appeared in [2]. In this work, we extend our work in [2] as follows. In addition to correlation analysis and predictive analysis, we conduct principal component analysis. This enables us to check the correlations between the original variables and the principal components, and allows us to find the proportions of variance for different principal components. A high proportion of variance for a principal component indicates that it can explain most of the variance in the dataset. In addition to using a single independent variable, we examine using a combination of independent variables to build the model using LR or RF for predicting the TCP throughput. We also investigate the use of principal components in generating RF models to predict the TCP throughput to indirectly confirm the hidden relationship between RTT and TCP throughput.

The rest of the paper is organized as follows. In addition to providing a brief overview of LR, RF, and PCA, Sect. 2 discusses the related work including prior works that have used LR and RF for prediction, and previous efforts on estimating wireline TCP throughput as well as on modelling and predicting throughput in WiFi networks. A brief description of the variables of interest in the WiFi dataset, namely link speed, received signal strength, round-trip time, number of available access points, and TCP throughput, along with a summary of different measures for these variables in this dataset is given in Sect. 3. Section 4 presents the details of the steps involved in carrying out the three different analyses including the procedure of splitting the dataset into a training set and a test set as well as summaries of the different measures for the variables in these sets. Results for the different analyses are provided in Sect. 5 along with an in-depth discussion. Conclusions are summarized in Sect. 6.

2 Background and related work

Machine learning techniques, like LR and RF, have been widely used in the literature for prediction [3–9]. Linear regression is used to estimate (or predict) values of a dependent variable by observing an independent variable. Random forest, on the other hand, has the ability to model highly nonlinear relationships. We use LR and RF on single or multiple independent variables to predict the dependent variable, i.e., TCP throughput in WiFi. Unlike other prediction-related works in the literature, our aim is not to come up with a precise prediction of the dependent variable; rather, we use prediction based on ML techniques, like LR and RF, as means to find any hidden relationship between other variables and throughput, specifically, to find a variable in the WiFi network that most significantly affects the throughput.

Principal component analysis is generally used to reduce the dimensionality of a dataset having a large number of interrelated variables while retaining as much as possible of the variation present in the dataset. This is done by generating a new set of uncorrelated variables referred to as the principal components. The first few PCs retain most of the variation present in all of the original variables [10]. In this work, we are not interested in using PCA to reduce the dimensionality of the WiFi dataset. Rather, we first study the correlations between the original variables and PCs as well as proportions of variance for PCs, and next, we use principal components in generating RF models to predict the TCP throughput to indirectly confirm any hidden relationship between other variables and throughput.

Previously, efforts have been made to investigate the issue of estimating wireline TCP throughput. Analytical models have been proposed for characterizing TCP throughput as a function of packet loss rate, round-trip time, and maximum segment size [11, 12]. Neural networks have been used to find patterns in the evolution of TCP throughput over time as a time series. These patterns are then used to predict the future TCP throughput over time [13]. To predict the estimated TCP throughput over time by using historical samples of observed throughput, prediction techniques based on genetic algorithms are applied to estimate the future TCP throughput values [14]. Support vector regression has been used for TCP throughput prediction that is based on prior file transfer history and measurements of simple path properties [15].

Modelling and predicting throughput in WiFi networks has been examined previously. Analytical prediction of system throughput has been conducted for WiFi networks based on medium access control parameters such as minimum and maximum backoff windows [16]. TCP throughput over rate-adaptive WiFi has also been analytically modelled [17]. Empirical models for throughput prediction of WiFi networks have been developed based upon throughput and signal-to-noise ratio measurements [18]. To predict the throughput, an empirical model has been proposed that best fits a simulated dataset for WiFi networks using directional antennas [19]. Environment-specific received signal strength measurements have been combined with simulations to predict the throughput performance of rate-adaptive WiFi [20]. Measurements of wireless client and access point interactions in WiFi have been used as input to a time series-based predictor of TCP throughput built on exponentially weighted moving average method [21].

Based on the analysis of traffic collected from WiFi testbeds, a statistical model that employs seasonal auto-regressive integrated moving average method has been proposed to predict the short-term traffic in WiFi networks [22]. From the collected throughput values, a statistical model that uses linear time series analysis based on auto-regressive method has been developed to predict the session throughput of constant bit-rate streams in WiFi [23]. Unlike these previous efforts on precise prediction of throughput, our work applies ML techniques on a publicly available WiFi dataset to find any hidden relationship between other variables and TCP throughput in WiFi and discovers a very close relationship between RTT and TCP throughput. The results also reveal that using RF in combination with PCA is extremely advantageous in finding hidden relationship between variables.

3 WiFi dataset

The WiFi dataset used in the three different types of analyses is a subset of the *Cell vs WiFi* dataset that is publicly available at http://web.mit.edu/cell-vs-wifi/downloads.html [1]. The *Cell vs WiFi* dataset was collected by Deng et al. using their publicly available *Cell vs WiFi* app [24]. This app collects packet-level traces for a 1 MB TCP upload and a 1 MB TCP download between the mobile device (or smartphone) and a server for both WiFi and cellular networks. The collected data is then uploaded to the server. We focus our investigation on the data for WiFi downlink in the WiFi dataset that we extracted from the downloaded *world-data.arff* dataset.

We adopted this WiFi dataset for our investigation due to the following reasons. It is a publicly available dataset that is easily accessible. The data in this dataset is crowd-sourced and it has been collected through a publicly available app. This dataset has been collected through the smartphones of users in several countries, which enables it to capture a wide variety of conditions.

Out of the variables available in the WiFi dataset, we consider the following five variables, and we refer to them as *variables of interest*:

- wifi_available_count
- wifi_rssi
- wifi_linkspeed
- wifi_tput
- wifi_rtt

wifi_available_count is the number of WiFi access points that are visible to the smartphone when its WiFi is turned on. wifi_rssi is the strength of the signal received at the smartphone in dBm of the WiFi access point that the smartphone is associated with. wifi_linkspeed is the data rate of the link in Mbps between the WiFi access point and the smartphone after the smartphone has associated with that access point. wifi_tput is the TCP throughput in Mbps and wifi_rtt is the TCP round-trip time in milliseconds, and they are measured by the *Cell vs WiFi* app on the smartphone in the downlink direction after the 1 MB TCP download between the smartphone and the server over the WiFi network and the Internet. A summary description of these variables along with their symbols that we adopt in this work is given in Table 1. The number of available WiFi access points, received signal strength, data rate of the WiFi link, and RTT are the variables that can significantly affect the throughput and we aim to use ML techniques to find the variable that most significantly affects the TCP throughput in WiFi.

Table 2 presents a summary of the different measures for the variables of interest in the entire WiFi dataset. These measures include the minimum, first quantile, median, mean, third quantile, and maximum values for a variable. It is interesting to note that S varies from 389 bps to 22.9 Mbps while T varies from 4.5 ms to 2.08 s.

Table 1 Description of variables of interest

Variable	Description	Symbol
wifi_available_count	Number of available WiFi access points	M
wifi_rssi	Received signal strength indicator in dBm	I
wifi_linkspeed	Link speed (or link data rate) in Mbps	R
wifi_tput	TCP throughput in Mbps	S
wifi_rtt	TCP round-trip time in ms	T

Table 2 Summary of variables of interest in WiFi downlink

Measure	*M*	*I* (dBm)	*R* (Mbps)	*S* (Mbps)	*T* (ms)
Minimum	4.00	− 86.00	1.00	0.000389	4.577
1st Quantile	13.00	− 70.00	6.00	0.746936	17.334
Median	25.00	− 65.00	26.00	1.828164	33.813
Mean	22.88	− 64.69	32.81	3.333310	87.916
3rd Quantile	30.00	− 59.00	54.00	4.601842	78.216
Maximum	40.00	− 35.00	65.00	22.950721	2085.083

4 Methodology for analysis

In this section, we describe the methodology for the three different types of analyses that we employ to investigate any hidden relationship between other variables and TCP throughput in WiFi. We apply these different analyses on the WiFi dataset to specifically find the variable that most significantly affects S (or TCP throughput in WiFi).

To check for visible correlation between other variables and S via correlation analysis, we generate pairwise scatter plots with linear regression fits, and pairwise correlations between variables of interest. To study the weights of variables of interest for the principal components as well as the proportions of variance for the principal components, we generate principal components using principal component analysis. To generate principal components, we use the *prcomp* function [25] in the *stats* package [26] in R. The *weights*, also known as *loadings*, are the correlation coefficients between the original variables and the PCs, and are used to calculate the principal component scores. The *proportion of variance* for a PC indicates its ability to account for the variance in the dataset. For example, a proportion of variance of 0.75 for a PC means that it can explain 75% of the total variance in the dataset.

To find any hidden relationship between other variables and S via predictive analysis, we employ well-known machine learning techniques such as linear regression and random forest. In this work, we use the term *linear regression* to represent cases where either single or multiple independent variables have been used. Random forest is an ensemble method that combines the result of several decision trees. It was developed as an extension to classification and regression trees approach [27] and consists of many decision trees, i.e., a forest, where each tree is generated from a new training dataset, which is a subset that is sampled randomly from the original training dataset. The tree splitting process is based on a subset of variables selected randomly from the set of all independent variables. Once all trees are developed, the final result of this ML technique would be the mean of the results of all trees. The parameter n_{tree} is the number of trees in the forest and we set it to 100 in this work.

For predictive analysis, we divide the dataset into two subsets, the training set and the test set. As per common practice in the literature, we use random sampling without replacement with an 80–20 ratio to separate the dataset into training and test sets [28]. We split the dataset such that 80% of the data in the WiFi dataset that is randomly selected comprises the training set while the remaining 20% constitutes the test set.

Tables 3 and 4 provide a summary of the different measures for the variables of interest in training set and test set, respectively, including their minimum, first quantile, median,

Table 3 Summary of variables of interest in training set

Measure	*M*	*I* (dBm)	*R* (Mbps)	*S* (Mbps)	*T* (ms)
Minimum	4.00	− 86.00	1.00	0.000389	4.577
1st Quantile	13.00	− 70.00	6.00	0.749594	17.182
Median	25.00	− 65.00	26.00	1.829997	33.691
Mean	22.86	− 64.68	32.85	3.338773	87.521
3rd Quantile	30.00	− 59.00	54.00	4.605131	78.216
Maximum	40.00	− 35.00	65.00	22.950721	2085.083

Table 4 Summary of variables of interest in test set

Measure	*M*	*I* (dBm)	*R* (Mbps)	*S* (Mbps)	*T* (ms)
Minimum	4.00	− 86.00	1.00	0.002534	4.577
1st Quantile	13.00	− 71.00	6.00	0.738652	17.456
Median	25.00	− 65.00	26.00	1.817592	34.119
Mean	22.95	− 64.72	32.68	3.311456	89.499
3rd Quantile	30.00	− 59.00	54.00	4.580232	79.559
Maximum	40.00	− 35.00	65.00	22.950721	2085.083

mean, third quantile, and maximum values. In the training set, S varies from 389 bps to 22.9 Mbps and T varies from 4.5 ms to 2.08 s as observed from Table 3. Furthermore, Table 4 shows that S varies from 2,534 bps to 22.9 Mbps and T varies from 4.5 ms to 2.08 s in the test set.

Once we have the training set and the test set, we use variables in the training set as input to LR or RF to generate the fitted (or trained) model. We use the *lm* function [29] in the *stats* package in R, and the *randomForest* function [30] in the *randomForest* package [31] in R to generate the model using LR and RF, respectively. Next, we use the fitted model along with variables in the test set to predict S in the test set, and we use the *predict* function [32] in the *stats* package in R to generate predicted values for S. For model fitting, we either use a single variable in the training set, such as M, I, R, or T, or multiple (two or more) variables for training on S in the training set, and for prediction, we use corresponding single or multiple variables in the test set.

To indirectly confirm the hidden relationship between other variables and S via predictive analysis, we use principal components in generating RF models. First, we add the principal component score vectors for the principal components generated during PCA to the dataset as new columns, then split the dataset into training and test sets according to the procedure explained earlier, and lastly, we use a principal component—more precisely, the column in the training set corresponding to the score vector of that PC—for training on S using RF. Next, we use the trained model along with the PC in the test set to predict S.

We selected linear regression for predictive analysis in this work as it is an elementary yet efficient ML technique that is widely used to study linear relationships among variables. We chose random forest as it is a well-known ML method that has been successfully used throughout the years to model highly nonlinear relationships. Instead of using principal component analysis for its common purpose of reducing the dimensionality of a dataset, we adopted PCA for its ability to provide: (1) correlations between the original variables and principal components, and (2) proportions of variance for the principal components. In addition, we used the score vectors for principal components from PCA in generating RF models to predict the TCP throughput. This enabled us to indirectly confirm any hidden relationship between other variables and throughput.

5 Results for analysis

The results for the three different types of analyses on the WiFi dataset, namely correlation analysis, principal component analysis, and predictive analysis, to investigate any hidden relationship between other variables and TCP throughput are presented here followed by an in-depth discussion at the end of this section.

5.1 Correlation analysis

Figure 1 illustrates pairwise scatter plots with linear regression fits, pairwise correlations, and histograms of variables of interest. In this figure, the pairwise or bivariate scatter plots with linear regression fits are shown below the diagonal; the histograms of the variables are shown on the diagonal; and correlation coefficients between a pair of variables are shown above the diagonal.

As can be seen from this figure, strong positive correlation is present between *R* (or link speed) and *I* (or received signal strength indicator). This is expected as received signal strength affects the signal-to-interference-plus-noise ratio, which determines the link speed or link data rate. Significant positive correlation is observed between *S* (or TCP throughput in WiFi) and *R*, and between *S* and *I*. This also makes sense as throughput directly depends upon the link speed and indirectly depends upon the received signal strength. Relatively low negative correlation is seen between *S* and *T* (or round-trip time), and between *S* and *M* (or number of available WiFi access points). This is also understandable as a higher RTT or a higher number of available access points (or

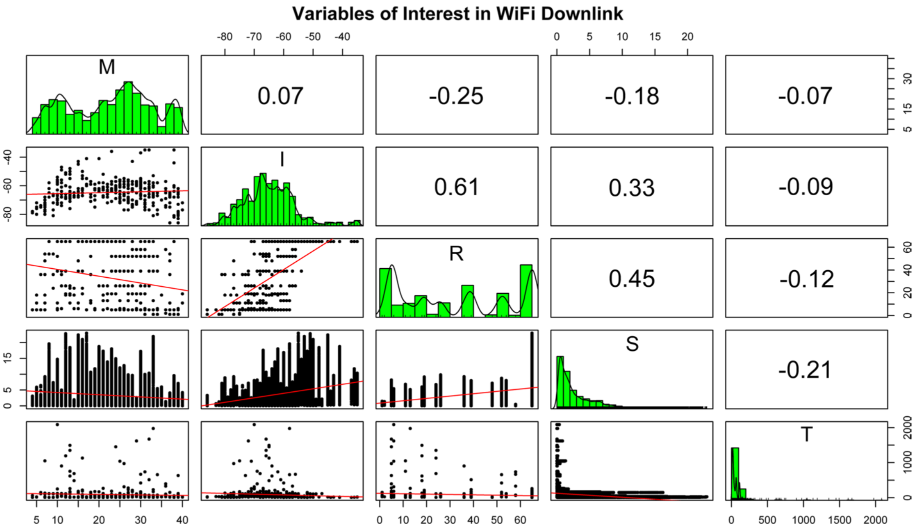

Fig. 1 Pairwise scatter plots with linear regression fits, pairwise correlations, and histograms of variables of interest in WiFi downlink

Table 5 Weights of variables of interest for principal components

Variable	PC1	PC2	PC3	PC4	PC5
M	0.00308	0.11650	− 0.92561	− 0.35626	− 0.05244
I	0.00381	− 0.23621	− 0.37716	0.89317	0.06470
R	0.01337	− 0.96199	− 0.02112	− 0.26710	0.05114
S	0.00377	− 0.07089	0.02318	0.06312	− 0.99521
T	− 0.99989	− 0.01367	− 0.00449	− 0.00102	− 0.00298

Table 6 Proportions of variance for principal components

Principal component	Proportion of variance
PC1	0.98380
PC2	0.01294
PC3	0.00216
PC4	0.00085
PC5	0.00024

interferers) would negatively impact the throughput. Based on these visible correlations, one can presume that R or I or their combination with other variables may act as reasonable predictors of S. On the other hand, one may not expect T or M alone to be reasonable predictors of S based on simple correlation analysis. Note that the pairwise scatter plot between S and T in Fig. 1 indicates a nonlinear relationship between the two.

5.2 Principal component analysis

After applying PCA on the variables of interest, the weights (also known as loadings) of variables of interest for the principal components are given in Table 5. They represent positive or negative correlations between the original variables and the principal components. It can be seen from this table that the weights of all variables for the first principal component, i.e., PC1, are small except for the weight of T. PC1 appears to be highly correlated to the variable T. Similarly, the *fifth principal component* (PC5) turns out to be highly correlated to the variable S. Table 6 shows the proportions of variance for the principal components. Note that the proportion of variance for PC1 is 0.9838 in this table. This means that PC1 can explain 98% of the total variance in the dataset, which means that nearly all of the information in the dataset can be encapsulated by the first principal component.

5.3 Predictive analysis

In predictive analysis, we compare the performance of different variables or their combinations or different PCs—that are employed by the machine learning techniques to predict the throughput—in terms of percentage of variance and root mean square error. PoV is the percentage of total variance explained in the dependent variable in the training set by the independent variable(s) in the model that is constructed using a machine learning technique. In our case, the dependent variable is S and the independent variables can be

M, I, R, and T or PC1, PC2, PC3, PC4, and PC5. PoV is based on the *coefficient of determination* (also known as *coefficient of multiple determination*) [33] and is given by

$$\mathrm{PoV} = \left(1 - \frac{\sum_{i=1}^{N}(A_i - F_i)^2}{\sum_{i=1}^{N}\left(A_i - \overline{A}\right)^2}\right) \times 100, \tag{1}$$

where A_i is the ith actual value of S in the training set, F_i is the corresponding ith fitted value of S that is generated by the machine learning model, N is the number of observations for S in the training set, and $\overline{A}$ is the mean of actual values of S in the training set.

RMSE between predicted values and actual values [34] of S in the test set is given by

$$\mathrm{RMSE} = \sqrt{\frac{1}{N'}\sum_{i=1}^{N'}\left(A_i' - P_i'\right)^2}, \tag{2}$$

where $A_i^{'}$ is the ith actual value of S in the test set, $P_i^{'}$ is the corresponding ith predicted value of S, and $N^{'}$ is the number of observations for S in the test set. Higher PoV and lower RMSE indicate better performance.

5.3.1 Linear regression

Table 7 shows the percentage of variance explained in S in the training set by one or more independent variables when used in generating the model based on linear regression. A very low PoV of 4.27% is observed when T alone is used in generating fitted values for S by this model. The PoV improves when multiple independent variables are used for model fitting, and a PoV of 24.49% is obtained when R, I, T, and M are used together.

RMSE between actual values of S in the test set and predicted values is given in Table 7 as well. A RMSE of 3.71 is seen when T in the test set is used in combination with the fitted LR model to generate the predicted values for S. The RMSE decreases when multiple independent variables (or predictors) in the test set are used to generate the predicted values, and a RMSE of 3.31 is achieved when R, I, T, and M are used together for prediction.

Figures 2 and 3 illustrate actual values of S versus predictions generated when using the LR model in combination with a single predictor and multiple predictors, respectively. The black coloured line, referred to as *S in test set* in these figures, represents the actual values of S in the test set. It is clear from these figures that the quality of prediction

Table 7 PoV and RMSE for independent variable(s) with LR model

ML technique	Independent variable(s) used in model fitting and prediction	PoV	RMSE
Linear regression	R	20.73	3.39
	I	10.65	3.59
	T	**4.27**	**3.71**
	M	3.16	3.74
	$R+I$	21.10	3.38
	$R+I+T$	23.44	3.33
	$R+T+M$	23.76	3.33
	$R+I+T+M$	**24.49**	**3.31**

Important results are highlighted in bold

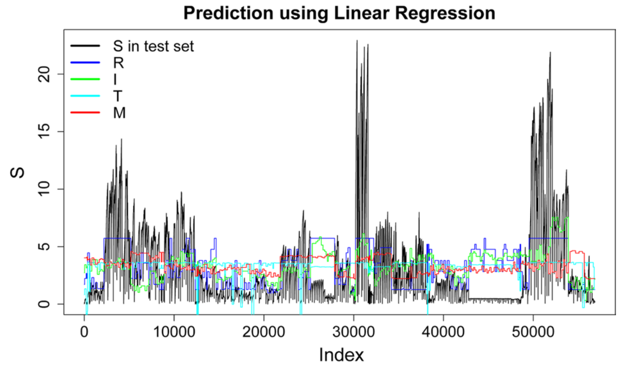

Fig. 2 Actual versus predicted values of *S* when using LR model and a single predictor

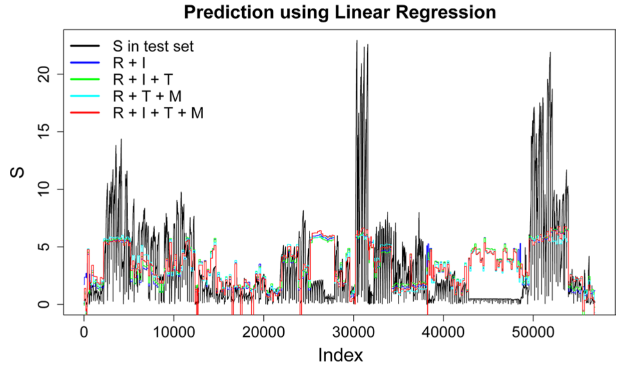

Fig. 3 Actual versus predicted values of *S* when using LR model and multiple predictors

Table 8 PoV and RMSE for independent variable(s) with RF model

ML technique	Independent variable(s) used in model fitting and prediction	PoV	RMSE
Random forest	R	34.29	3.08
	I	32.89	3.10
	T	**87.63**	**1.33**
	M	18.13	3.44
	$R+I$	47.03	2.76
	$R+I+T$	76.88	1.81
	$R+T+M$	76.84	1.82
	$R+I+T+M$	**83.45**	**1.53**

Important results are highlighted in bold

based on the model that uses linear regression is poor as the predictions are far from the actual values of *S*. This poor performance is due to the fact that the LR model could only capture up to 24.49% of the variation in *S*.

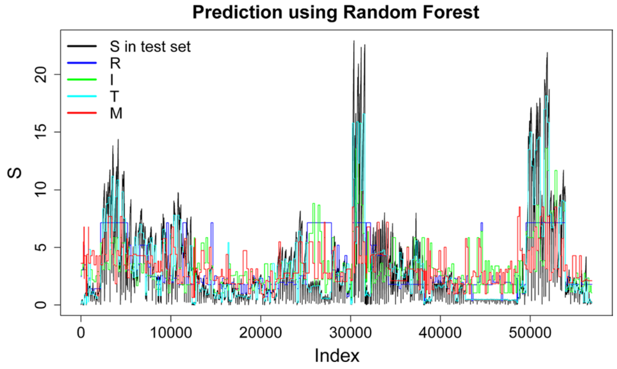

Fig. 4 Actual versus predicted values of *S* when using RF model and a single predictor

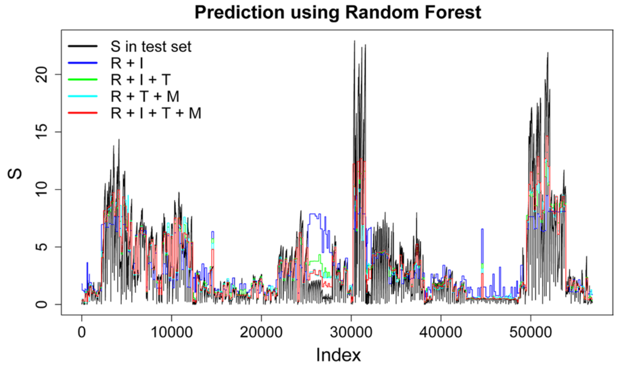

Fig. 5 Actual versus predicted values of *S* when using RF model and multiple predictors

5.3.2 Random forest

The percentage of variance explained in S in the training set by single or multiple independent variables with the model generated using random forest is shown in Table 8. As seen earlier in case of LR model, the PoV with RF model improves when multiple independent variables are used for model fitting. The PoV is 83.45% when R, I, T, and M are used together. However, it should be noted that this PoV is lower than that obtained when only using T for model fitting. A PoV of 87.63% is observed when T is used in generating fitted values for S, which is also much higher than a PoV of 4.27% seen with T in the LR model.

Table 8 also reveals RMSE when a single predictor or multiple predictors in the test set are used in combination with the fitted RF model to generate predicted values for S. The RMSE decreases with an increasing number of predictors, and when R, I, T, and M are used together as predictors, a RMSE of 1.53 is seen. However, an even lower RMSE of 1.33 is achieved when only using T for prediction.

Figures 4 and 5 show actual versus predicted values of S generated when using the model based on RF with a single predictor and multiple predictors, respectively. When using a single variable for prediction, T outperforms others as is clearly visible in Fig. 4. The quality of prediction improves with an increasing number of predictors as noticed

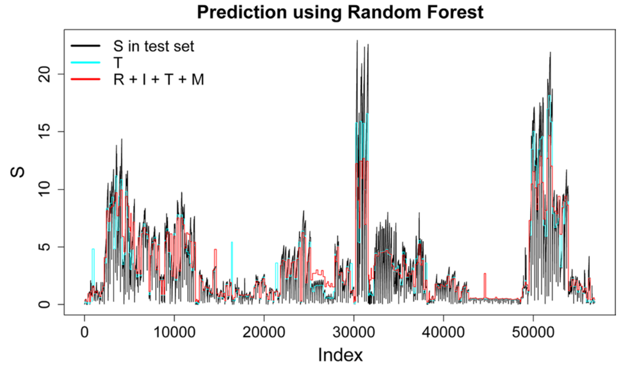

Fig. 6 Actual versus predicted values of *S* based on RF model—*T* versus all predictors

Table 9 PoV and RMSE for a principal component with RF model

ML technique	Principal component used in model fitting and prediction	PoV	RMSE
Random forest	PC1	**99.61**	**0.23**
	PC2	87.30	1.37
	PC3	90.95	1.17
	PC4	80.34	1.67
	PC5	91.53	1.11

Important results are highlighted in bold

from Fig. 5. However, *T* outperforms the combination of all predictors as observed from Fig. 6. When used in model fitting that employs random forest, *T* is able to capture a very high percentage of variation in *S*, which results in good performance.

As can be seen from Tables 7 and 8 as well as Figs. 3 and 5, the combination of *R*, *I*, and *T* and that of *R*, *T*, and *M* perform similarly. This indicates that exhaustively exploring all combinations of different independent variables for predicting *S* is not necessary, and we employ only a few combinations consisting of two or three or all four independent variables to examine their performance in comparison with only using a single independent variable to predict *S*.

5.3.3 Random forest with principal components

When a principal component is used in combination with RF for model fitting and prediction, the PoV explained in *S* in the training set as well as the RMSE between actual values of *S* in the test set and predicted values are shown in Table 9. A very high PoV of 99.61% is achieved when PC1 is used in generating fitted values with the RF model. Also, a very low RMSE of 0.23 is observed when PC1 is used for prediction. This is reflected in Fig. 7, which shows actual versus predicted values of *S*. It is also observed from Table 9 that PC1 performs better than PC5 in terms of PoV and RMSE.

Recall that PC1 was found to be highly correlated to *T* and was able to explain 98% of the total variance in the WiFi dataset as was highlighted in Tables 5 and 6. Also, PC5 had been found to be highly correlated to *S* during PCA. Previously, we found that when *T*

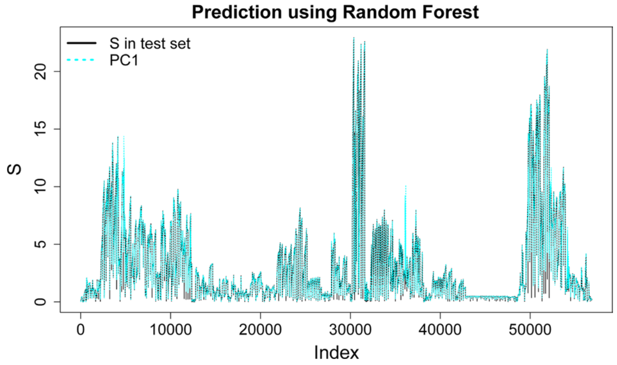

Fig. 7 Actual versus predicted values of *S* when using RF model and PC1

was used for generating the fitted model using RF, it outperformed other single and multiple predictors. Currently, we observe that PC1—which was earlier found to be highly correlated to *T*—provides an excellent performance, thereby indirectly re-confirming the ability of *T* to closely predict *S*. The need to use more computationally complex ML techniques, like neural networks, to achieve higher accuracy did not arise as a very high prediction accuracy was achieved with RF.

5.4 Discussion

As mentioned in Section 3, the crowd-sourced WiFi dataset, which we have employed for these different analyses, was collected using the publicly available *Cell vs WiFi* app [24]. This app gathers packet-level traces between a smartphone (located anywhere in the world) and a server (located at the Massachusetts Institute of Technology). If WiFi is available and the smartphone is connected to the Internet, the app, when activated by the smartphone user, initiates a 1 MB TCP download at the smartphone from the server over the WiFi network and the Internet, and collects packet-level measurements for *S* and *T* among other variables. The app then uploads the collected data along with the smartphone's geographic location to the server.

RTT (or *T*) consists of the time it takes a packet to reach the smartphone from the server and the corresponding acknowledgement to reach the server from the smartphone. This includes all different types of delays encountered over the WiFi link and the intermediate wired links along both paths (i.e., from server to smartphone and from smartphone to server) such as transmission delay (which includes delay caused by MAC layer-level retransmissions due to problems over the WiFi link), propagation delay, queueing delay, and processing delay.

Congestion is the primary concern for TCP. It happens when routers are overwhelmed with traffic. This causes their queues to build up, which eventually overflow causing packet losses that lead to delays due to packet retransmissions. TCP is unable to differentiate between WiFi-link losses and wired-link losses, and assumes that all packet losses indicate congestion. When TCP detects a loss, it retransmits the lost packet, and in addition it reduces its transmission rate [35]. This relieves congestion by draining router queues. Packet losses over WiFi can occur due to wireless channel errors;

collisions while accessing the medium at the MAC layer level; and/or buffer overflow at an access point [36]. A packet loss over WiFi is resolved by a TCP-level retransmission.

TCP needs to periodically measure RTT to set the value of its *retransmission timeout* (RTO) [37]. The value of RTO is set slightly higher than RTT. If RTT increases, RTO increases, and the sender has to wait longer to retransmit the lost packet, which negatively impacts throughput. Moreover, when RTO expires, TCP sees this packet loss as congestion and reduces its congestion window, which degrades throughput. On the other hand, if RTT decreases, TCP continues to operate in its current phase and keeps increasing its congestion window, which improves throughput. This indicates an inverse relationship between RTT and throughput, and generally the TCP throughput is given by $cwnd/RTT$, where *cwnd* is the size of TCP's congestion window [38].

From correlation analysis, we observed a significant positive correlation between throughput (or S) and link speed (or R), and between throughput and received signal strength (or I). Relatively low negative correlation was seen between throughput and RTT (or T), and between throughput and number of available access points (or M). Based on visible correlations, one could presume that a combination of multiple variables, including R and I, would act as reasonable predictor of S, whereas T or M alone would not be expected to reasonably predict S. The principal component analysis revealed that the first principal component (or PC1) was highly correlated to T, whereas PC5 was highly correlated to S. The proportion of variance for PC1 turned out to be 0.9838 during PCA, implying that PC1 could explain 98% of the total variance in the dataset.

To compare performance of different variables or their combinations or different PCs that were employed by the machine learning techniques to predict S, percentage of variance was used to measure the quality of the fitted model whereas root mean square error was used to measure the accuracy of prediction resulting from the fitted model. A maximum of 24.49% of the variation in S was captured when all variables, including R, I, T, and M, were used for generating fitted values by the model that employed linear regression, and this low PoV resulted in poor quality of prediction. Also, a very low PoV of 4.27% was observed when only T was used in generating fitted values with the LR model.

When used in generating fitted and predicted values for S with the model that employed random forest, T was able to achieve a PoV of 87.63%, a RMSE of 1.33, and outperformed all single variables as well as their combinations by providing the highest PoV and the lowest RMSE. A very high PoV of 99.61% and a very low RMSE of 0.23 were observed when PC1 was used in combination with the RF model to generate fitted and predicted values, respectively. PC1 even performed better than PC5 in terms of PoV and RMSE. Recall that PC5 was earlier found to be highly correlated to S.

Previously, TCP throughput has been modelled as a function of different variables including RTT [11, 12]. The relationship between TCP throughput and measurements of path properties, including queueing delays and packet loss, has been investigated using machine learning [15]. It is shown that TCP throughput predictions can be improved by up to a factor of 3 when these path properties are considered by a support vector regression-based machine learning model. While investigating hidden relationships among variables in WiFi using random forest-based machine learning models, we discover a very significant relationship between S and T. In fact, our investigation using machine learning reveals RTT as the variable that most significantly affects TCP throughput in WiFi.

6 Conclusions

In this work, we studied the relationship between TCP throughput and other variables in the WiFi network such as link speed, received signal strength, RTT, and number of available WiFi access points. The objective was to discover any hidden relationship between throughput and these variables using machine learning. More specifically, we were interested in discovering a variable in the WiFi network that significantly affected throughput. To this end, we conducted three different types of analyses on a publicly available WiFi dataset, which included employing ML techniques like linear regression, random forest, and principal component analysis.

A correlation analysis was not conclusive about the effect of other variables on TCP throughput in WiFi. The LR model was not able to find the hidden relationship between throughput and RTT as linear regression is only useful for capturing linear relationships between variables. With the RF model, RTT succeeded in closely capturing the variation in throughput due to random forest's ability to capture highly nonlinear relationships between variables. The first principal component, that had been found to be highly correlated to RTT during principal component analysis, captured almost all of the variation in throughput when used with the RF model. This indirectly re-confirmed the ability of RTT to closely predict TCP throughput.

TCP throughput has an inverse relationship with RTT, and it has previously been modelled as a function of RTT among other variables. Unlike any previous effort on modelling and predicting TCP throughput, our investigation reveals RTT as the sole variable that most accurately predicts and thereby most significantly affects TCP throughput in WiFi. Our work discovers a very close relationship between RTT and TCP throughput, and these results can be beneficial in advancing the knowledge of the relationship between latency and throughput for designing better low-latency networks.

Abbreviations

LR: Linear regression; ML: Machine learning; PC: Principal component; PC1: First principal component; PC5: Fifth principal component; PCA: Principal component analysis; PoV: Percentage of variance; RF: Random forest; RMSE: Root mean square error; RTO: Retransmission timeout; RTT: Round-trip time; ULL: Ultra-low latency.

Authors' contributions

AUC is the sole author of this paper and carried out all aspects of the work in this study. The author read and approved the final manuscript.

Funding

Not applicable.

Availability of data and materials

The dataset employed for conducting the current study is publicly available at http://web.mit.edu/cell-vs-wifi/downloads.html.

Declarations

Competing interests

The authors declare that they have no competing interests.

Received: 21 May 2021 Accepted: 7 December 2021

References

1. S. Deng et al., *Cell vs WiFi, Electronic Dataset*. Available: http://web.mit.edu/cell-vs-wifi/downloads.html
2. A.U. Chaudhry, R.H.M. Hafez, On Finding Hidden Relationship among Variables in WiFi using Machine Learning. in Proceedings. *2020 International Conference on Computing, Networking and Communications (ICNC '20) Workshop on Computing, Networking and Communications (CNC '20)*, Big Island, Hawaii, USA, 2020
3. C. Chagas, W. Junior, S. Bhering, B. Filho, Spatial prediction of soil surface texture in a semiarid region using random forest and multiple linear regressions. CATENA **139**, 232–240 (2016)
4. L. Candanedo, V. Feldheim, D. Deramaix, Data driven prediction models of energy use of appliances in a low-energy house. Energy Build **140**, 81–97 (2017)
5. C. Lei, J. Deng, K. Cao, L. Ma, Y. Xiao, L. Ren, A random forest approach for predicting coal spontaneous combustion. Fuel **223**, 63–73 (2018)
6. I. Laory, T. Trinh, I. Smith, J. Brownjohn, Methodologies for predicting natural frequency variation of a suspension bridge. Eng. Struct. **80**, 211–221 (2014)
7. P. Smith, S. Ganesh, P. Liu, A comparison of random forest regression and multiple linear regression for prediction in neuroscience. J. Neurosci. Methods **220**(1), 85–91 (2013)
8. A. Knudby, A. Brenning, E. LeDrew, New approaches to modelling fish-habitat relationships. Ecol. Model. **221**(3), 503–511 (2010)
9. A. Chlingaryan, S. Sukkarieh, B. Whelan, Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: a review. Comput. Electron. Agric. **151**, 61–69 (2018)
10. I.T. Jolliffe, *Principal Component Analysis*, 2nd edn. (Springer, New York, 2002)
11. M. Mathis, J. Semke, J. Mahdavi, T. Ott, The macroscopic behavior of the TCP congestion avoidance algorithm. Comput. Commun. Rev. **27**(3), 67–82 (1997)
12. J. Padhye, V. Firoiu, D. Towsley, J. Kurose, Modeling TCP throughput: a simple model and its empirical validation. Comput. Commun. Rev. **28**(4), 303–314 (1998)
13. P. Cortez, M. Rio, M. Rocha, P. Sousa, Multi-scale internet traffic forecasting using neural networks and time series methods. Expert. Syst. **29**(2), 143–155 (2012)
14. C. Benet, A. Kassler, E. Zola, Predicting expected TCP throughput using genetic algorithm. Comput. Netw. **108**, 307–322 (2016)
15. M. Mirza, J. Sommers, P. Barford, X. Zhu, A machine learning approach to TCP throughput prediction. IEEE/ACM Trans. Network. **18**(4), 1026–1039 (2010)
16. G. Bianchi, Performance analysis of the IEEE 802.11 distributed coordination function. IEEE J. Sel. Areas Commun. **18**(3), 535–547 (2000)
17. C. Burmeister, U. Killat, J. Bachmann, TCP over Rate-Adaptive WLAN—An Analytical Model and its Simulative Verification. in Proceedings. *2006 International Symposium on a World of Wireless, Mobile and Multimedia Networks (WOWMOM '06)*, Washington, USA, 2006
18. C. Na, J. Chen, T. Rappaport, Measured traffic statistics and throughput of IEEE 802.11b public WLAN hotspots with three different applications. IEEE Trans. Wirel. Commun. **5**(11), 3296–3305 (2006)
19. S. Kandasamy, R. Morla, P. Ramos, M. Ricardo, Predicting throughput in IEEE 802.11 based wireless networks using directional antenna. Wirel. Netw. **25**, 1567–1584 (2019)
20. P. Gopalakrishnan, P. Spasojevic, L. Greenstein, I. Seskar, A Method for Predicting the Throughput Characteristics of Rate-Adaptive Wireless LANs. in Proceedings. *IEEE 60th Vehicular Technology Conference (VTC '04-Fall)*, Los Angeles, USA, 2004
21. M. Mirza, K. Springborn, S. Banerjee, P. Barford, M. Blodgett, X. Zhu, On the Accuracy of TCP Throughput Prediction for Opportunistic Wireless Networks. in Proceedings. *6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '09)*, Rome, Italy, 2009
22. C. Chen, Q. Pei, L. Ning, Forecasting 802.11 Traffic using Seasonal ARIMA Model. in Proceedings. *2009 International Forum on Computer Science-Technology and Applications (IFCSTA '09)*, Chongqing, China, 2009
23. L. Cheng, I. Marsic, Modeling and Prediction of Session Throughput of Constant Bit Rate Streams in Wireless Data Networks. in Proceedings. *2003 IEEE Wireless Communications and Networking Conference (WCNC '03)*, New Orleans, USA, 2003
24. S. Deng, R. Netravali, A. Sivaraman, H. Balakrishnan, WiFi, LTE, or Both? Measuring Multi-Homed Wireless Internet Performance. in Proceedings. *2014 Internet Measurement Conference (IMC '14)*, Vancouver, Canada, 2014
25. R.A. Becker, J.M. Chambers, A.R. Wilks, *The New S Language* (Wadsworth & Brooks/Cole, California, 1988)
26. R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2018. Available: http://www.R-project.org/
27. L. Breiman, J. Friedman, C. Stone, R. Olshen, *Classification and Regression Trees* (Wadsworth & Brooks/Cole, California, 1984)
28. X. Amatriain, A. Jaimes, N. Oliver, J. Pujol, Data Mining Methods for Recommender Systems. in *Recommender Systems Handbook* (Springer, New York, 2010)
29. J.M. Chambers, T.J. Hastie, Chapter 4: Linear Models. in *Statistical Models in S* (Wadsworth & Brooks/Cole, California, 1992)
30. L. Breiman, Random forests. Mach. Learn. **45**, 5–32 (2001)
31. L. Breiman et al., *Package "randomForest"—Breiman and Cutler's Random Forests for Classification and Regression*. Available: https://cran.r-project.org/web/packages/randomForest/randomForest.pdf
32. J.M. Chambers, T.J. Hastie, Chapter 6: Generalized Linear Models. in *Statistical Models in S* (Wadsworth & Brooks/Cole, California, 1992)
33. D. Montgomery, *Design and Analysis of Experiments* (Wiley, New York, 1991)
34. J. Esfahani et al., Comparison of experimental data, modelling and non-linear regression on transport properties of mineral oil based nanofluids. Powder Technol. **317**, 458–470 (2017)
35. G. Xylomenos, G.C. Polyzos, P. Mahonen, M. Saaranen, TCP performance issues over wireless links. IEEE Commun. Mag. **39**(4), 52–58 (2001)

36. S.R. Pokhrel, M. Panda, H.L. Vu, M. Mandjes, TCP performance over Wi-Fi: joint impact of buffer and channel losses. IEEE Trans. Mob. Comput. **15**(5), 1279–1291 (2016)
37. B.A.A. Nunes, K. Veenstra, W. Ballenthin, S. Lukin, K. Obraczka, A Machine Learning Approach to End-to-End RTT Estimation and its Application to TCP. in Proceedings. *20th International Conference on Computer Communications and Networks (ICCCN '11)*, Maui, Hawaii, USA, 2011
38. C.P. Fu, S.C. Liew, TCP veno: TCP enhancement for transmission over wireless access networks. IEEE J. Select. Areas Commun. **21**(2), 216–228 (2003)

Publisher's Note