ÇUKUROVA UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES

MSc THESIS

Yasin YUR

VOICE AND DATA TRAFFIC MODELING AND PREDICTION FOR A THIRD GENERATION MOBILE NETWORK USING MACHINE LEARNING METHODS

DEPARTMENT OF COMPUTER ENGINEERING

ADANA, 2017

ÇUKUROVA UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES

VOICE AND DATA TRAFFIC MODELING AND PREDICTION FOR A THIRD GENERATION MOBILE NETWORK USING MACHINE LEARNING METHODS

Yasin YUR

MSc THESIS

DEPARTMENT OF COMPUTER ENGINEERING

We certify that the thesis titled above was received and approved the award of degree of the Master of Science by the board of jury on

……...	....	....
Assoc. Prof. Dr. M.Fatih AKAY	Asst. Prof. Dr. B.Melis ÖZYILDIRIM	Asst. Prof. Dr. Onur ÜLGEN
SUPERVISOR	MEMBER	MEMBER

This MSc Thesis is written at the Department of Institute of Natural And Applied Sciences of Çukurova University.
Registration Number:

Prof. Dr. Mustafa GÖK
Director
Institude of Natural and Applied Science

This study was supported by Research Projects Unit Ç.U. Project Number: FYL-2016-6403

ABSTRACT

MASTER THESIS

VOICE AND DATA TRAFFIC MODELING AND PREDICTION FOR A THIRD GENERATION MOBILE NETWORK USING MACHINE LEARNING METHODS

Yasin YUR

ÇUKUROVA UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES
DEPARTMENT OF COMPUTER ENGINEERING

Supervisor	: Assoc. Prof. Dr. M. Fatih AKAY
	Year: 2017, Pages: 107
Jury	: Assoc. Prof. Dr. M. Fatih AKAY
	: Asst. Prof. Dr. B. Melis ÖZYILDIRIM
	: Asst. Prof. Dr. Onur ÜLGEN

The purpose of this thesis is to derive models for traffic characteristics of a 3G network which is commercially deployed in Turkey and predict voice and data traffic by using various machine learning methods. The machine learning methods which were employed are Support Vector Machines (SVM), Multilayer Perceptron (MLP), Random Forest (RF) and Radial Basis Function Neural Network (RBF). Additionally, the Holt-Winters method has been applied to develop prediction models as a statistical method. Four different type of UMTS network traffic data have been utilized in order to build traffic prediction models. The performance of the forecasting models for the data sets has been assessed using Mean Absolute Percentage Error (*MAPE*). Finally, the performance of statistical and machine learning regression methods have been compared and the results show that SVM and Holt-Winters based models usually perform better than the ones obtained by the other methods.

Key Words: Machine learning, time series, UMTS traffic forecasting, time lags

ÖZ

YÜKSEK LİSANS TEZİ

MAKİNE ÖĞRENME YÖNTEMLERİ KULLANILARAK ÜÇÜNCÜ NESİL MOBİL ŞEBEKELERDEKİ SES VE DATA TRAFİKLERİ İÇİN İLERİ YÖNLÜ TRAFİK TAHMİN MODELLERİNİN GELİŞTİRİLMESİ

Yasin YUR

ÇUKUROVA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

Danışman	: Doç. Dr. M. Fatih AKAY
	Yıl: 2017, Sayfa: 107
Jüri	: Doç. Dr. M. Fatih AKAY
	: Yrd. Doç. Dr. B. Melis ÖZYILDIRIM
	: Yrd. Doç. Dr. Onur ÜLGEN

Bu tezin amacı, üçüncü nesil mobil şebekelerin devre anahtarlamalı ses ve paket anahtarlamalı veri trafikleri için ileri yönlü tahminin yapılmasını sağlayan daha akıllı modeller geliştirmektir. Trafik tahmin modelleri ilk olarak makine öğrenimi ile oluşturulmuş ve içerik olarak dört farklı yöntem kullanılmıştır. Bu yöntemler sırası ile Destek Vektör Makinesi (Support Vector Machine - SVM), Çok Katmanlı Algılayıcı (Multilayer Perceptron - MLP), Radyal Tabanlı Fonksiyon Sinir Ağı (Radial Basis Function Neural Network – RBF) ve Rasgele Orman'dır (Random Forest). İlave olarak istatistiki bir yöntem olan Holt-Winters methodu kullanılarak trafik tahmin modeli geliştirilmiştir. Trafik tahmin modelleri üretilirken UMTS şebekesine ait olan dört farklı tipteki gerçek trafik bilgileri kullanılmıştır. Modellerin performansı Ortalama Mutlak Yüzde Hata (*MAPE*) değeri hesaplanarak değerlendirilmiştir. Sonuçlar, genel olarak SVM tabanlı ve Holt-Winters tabanlı modellerin diğer yöntemlerden daha iyi performans elde ettiğini göstermektedir.

Anahtar Kelimeler: Makine öğrenme, zaman serileri, UMTS trafik tahmini, zaman gecikmeleri

GENİŞLETİLMİŞ ÖZET

Günümüzde yeni nesil iletişim teknolojileri ve hava arayüzündeki artan frekans ve band genişlikleri ile daha fazla kapasitenin kullanıcıya sunulması, mobil ses ve data trafik talebini dramatik bir şekilde yükseltmektedir. Bununla beraber ülkemizde geçmiş zamanlara göre ucuzlayan mobil veri kullanımı ve farklı ihtiyaç paketlerinin kullanıcıya sunulması veri kullanımının hergeçen gün artmasına olanak sağlamaktadır. Bu gerçekler doğrultusunda artan trafik miktarının tahmin edilmesi işlemi mobil operatörler için şebeke kapasite analizi ile beraber yatırım gereksinimlerinin doğru boyutlandırılması ve dolayısı ile aboneye sunulacak servis kalitesinin arttırılması gerçekleri ile büyük önem arz etmektedir.

Üçüncü nesil mobil şebekeler için ileri yönlü trafik tahmin işlemini yapabilecek yöntemler ya da uygulamalar sınırlı sayıdadır. Ticari olarak satışı olan ürünlerin çeşitliliği ile ülke içerisinde bulunabilirliği yeterli seviyelerde değildir. Bunun yanında mevcut ve kullanımda olan geleneksel istatistikî yöntemlerde yer almaktadır. Ancak bu uygulamaların hata seviyeleri bu derece önemli ve kritik bir operasyon için çok yüksek seviyelerdedir. Bu gerekçelerden dolayı, trafik tahmin işlemini yapabilecek ve yapılan bu tahmini doğru ölçeklendirebilecek daha akıllı modellere büyük gereksinim duyulmaktadır.

Bu tezin amacı Türkiye'de ticari olarak servis veren bir operatörün üçüncü nesil mobil şebekesi için makine öğrenimine dayalı yöntemler kullanılarak ileri dönemli ses ve data trafik tahmini sağlayan modeller geliştimektir.

Literatürde trafik tahmini üzerine yapılmış çalışmalar mevcuttur. Bu çalışmalar içerisinde makine öğrenme yöntemleri ve istatiksel regresyon yöntemleri kullanılmıştır. Tez kapsamında, literatürdeki diğer tüm çalışmalardan farklı olarak; daha fazla sayıda ve daha önce denenmemiş olan makine öğrenme yöntemleri kullanılarak elde edilen modellerin performansları birbirleri ile ve ilave olarak istatiksel bir regresyon yöntemi olan Holt-Winters metodu kullanılarak oluşturulan modellerin performansıyla da karsılaştırılmıştır. Bu özgün değere ilave olarak litratürde yer alan diğer çalışmalar içerisinde Türkiye'deki üçüncü nesil mobil bir şebekenin trafik servis modellemesi çalışmasına rastlanmamıştır. Bu nedenle yüksek lisans tezi ile ilk defa Türkiye'de yer alan üçüncü nesil mobil bir şebekenin trafik modellemesi yapılmıştır. Ayrıca literatürdeki diğer çalışmaların içerisinde hem ses hemde data trafiğini aynı anda kapsayıp modelleyen bir çalışmada mevcut değildir. Yüksek lisans tezi bu iki trafik tipini aynı anda modelleyen ilk çalışma olmuştur.

Tez içerisinde üç farklı veri seti kullanılmıştır. Bu setler sırası ile saatlik, günlük ve haftalık olarak ses ve data trafik bilgilerini içermektedir. Ses trafiği veri seti içerisinde yer alan bilgilerin birimi Erlang ve data veri seti içerisindeki bilgilerin birimi ise bit olacaktır. Veri setleri Türkiye'de servis veren üçüncü nesil mobil bir şebekenin geçmiş dönemli canlı şebeke kayıtlarından derlenmiştir.

Makine öğrenme yöntemleri olarak Destek Vektör Makinesi (Support Vector Machine - SVM), Çok Katmanlı Algılayıcı (Multilayer Perceptron - MLP), Radyal Tabanlı Fonksiyon Sinir Ağı (Radial Basis Function Neural Network – RBF), Rasgele Orman (Random Forest) kullanılmıştır. İlave olarak istatiksel regresyon yöntemi kullanılarak da bir modelleme yapılmıştır. Bu işlem içerisinde yöntem olarak Holt-Winters metodu kullanılmıştır. Modellemeler sonucunda istatiksel regresyon yöntemi ve makine öğrenme yöntemlerinin performansları karsılaştırılmıştır. Sonuç olarak farklı makine öğrenme yöntemleri ve istatiksel regresyon yöntemleriyle analiz edilip, en iyi tahmin modelleri, en iyi tahmin modelini veren zaman gecikmeleri ve en başarılı yöntem karşılaştırmsı yapılmıştır. Modellerin performansı Ortalama Mutlak Yüzde Hata (Mean Absolute Percentage Error – MAPE) değeri hesaplanarak değerlendirilmiştir.

Elde edilen modellerin performansı göz önüne alındığında, çeşitli sonuçlara varılmaktadır. Her şeyden önce, SVM tabanlı modeller ve istatistiksel tabanlı Holt-Winters modelleri ile diğer regresyon yöntemleri kullanılarak geliştirilen modellere göre daha iyi tahmin sonuçları elde edilmiştir. Performans ölçütü MAPE'ye göre sonucu en iyiden en kötüye doğru sıralanacak şekilde 3G / UMTS şebekesi trafik tahmin performansı açısından regresyon yöntemlerinin başarı sırası, SVM, Holt-Winters, MLP, RF ve RBF olarak yer almaktadır.

Sonuç olarak bu tez genişletilerek gelecekteki çalışmalar için farklı alanlara taşınabilir. Farklı zaman gecikmeleri ile farklı makine öğrenme yöntemleri 3G / UMTS ağ trafiğini tahmin etmek için uygulanabilir. Ayrıca, bu çalışma kolayca dördüncü nesil (4G / LTE) gibi yeni nesil kablosuz telekomünikasyon teknolojisi yada yakın zamanda geliştirilmesi tamamlanarak uygulamaya dönüşecek beşinci nesil (5G) mobil ağlar için kullanılacak şekilde genişletilebilir.

ACKNOWLEDGEMENTS

Foremost, I would like to express my sincere gratitude to my advisor Assoc. Prof. Dr. M. Fatih AKAY, for his supervision guidance, encouragements, patience, motivation, useful suggestions and his valuable time for this work.

I would like to thank members of the MSc thesis jury, Asst. Prof. Dr. Buse Melis ÖZYILDIRIM and Asst. Prof. Dr. Onur ÜLGEN, for their suggestions and corrections.

I would also like to thank Cukurova University Scientific Research Projects Center for supporting this work (Project no: FYL-2016-6403).

Last but not the least, I would like to thank my wife and my family for their endless support and encouragements for my life and career.

CONTENTS **PAGE**

LIST OF TABLES **PAGE**

LIST OF FIGURES **PAGE**

LIST OF ABBREVIATIONS

ANFIS	:	Adaptive Neuro-Fuzzy Inference System
ANN	:	Artificial Neural Network
AR	:	Auto-Regressive
ARIMA	:	Auto-Regressive Integrated Moving Average
ARMA	:	Auto-Regressive Moving Average
BP	:	Back Propagation
BPNN	:	Back Propagation Neural Network
CS	:	Circuit Switch
CTSA	:	Chaotic Time Series Analysis
DHR	:	Dynamic Harmonic Regression
FARIMA	:	Auto-Regressive Fractionally Moving Average
FFNN	:	Feed-Forward Neural Network
FIRNN	:	Finite-Impulse-Response Neural Network
FIS	:	Fuzzy Inference System
FNT	:	Flexible Neural Tree
GA-RBF	:	Radial Basis Function optimized by Genetic Algorithm
GARCH	:	Generalized Auto-Regressive Conditional Heteroscedasticity
GM	:	Grey Model
GML	:	Gausian Maximum Likelihood
GP	:	Genetic Programming
HES	:	Holt-Exponential Smoothing
HM	:	Historical Mean
HSDPA	:	High-Speed Downlink Packet Access
HSUPA	:	High-Speed Uplink Packet Access
ISP	:	Internet Service Provider

KF	:	Kalman filtering
LA	:	Local Linear Approximation
LM	:	Levenberg-Marquardt
LR	:	Linear Regression
LS-SVM	:	Least Squares Support Vector Machine
LSVM	:	Local Support Vector Machine
M5P	:	Decision Tree with Linear Regression Functions at the Nodes
MA	:	Moving Average
MAPE	:	Mean Absolute Percentage Error
MLP	:	Multi-Layer Perceptrons
MMLP	:	Multiresolution Multi-Layer Perceptrons
MMPP	:	Markov-Modulated Poison Process
MODWT	:	Maximal Overlap Discrete Wavelet Transform
NN	:	Neural Network
NNB	:	Nearest Neighborhood
NNE	:	Neural Network Ensemble
OK	:	Ordinary Kriging
OL-SVR	:	Online Support Vector Regression
O&M	:	Operation and Maintenance
OOB	:	Out-Of-Bag
PS	:	Packet Switch
RBF	:	Radial Basis Function Neural Network
REPTree	:	Reduced Error Pruning
RF	:	Random Forest
RNC	:	Radio Network Controller
RP	:	Resilient Back Propagation
RT	:	Random Tree

SAE	:	Stacked Auto encoder
SARIMA	:	Seasonal Autoregressive Moving Average
SHW	:	Seasonal Holt-Winters
SPN	:	Spinning Network
SSVRCSA	:	Seasonal SVR with Chaotic Annealing Algorithm
SVM	:	Support Vector Machines
SVR	:	Support Vector Regression
UMTS	:	Universal Mobile Telecommunication System
WCDMA	:	Wideband Code Division Multiple Access
3GPP	:	3rd Generation Partnership Project

1. INTRODUCTION

1.1. UMTS Network

Since 1998, Third Generation Mobile Telecommunication System has been deployed for enabling mobile accessible wireless data (packet switched) and voice (circuit switched) services. Universal Mobile Telecommunication System (UMTS) is the term used in Europe for 3rd generation (3G) networks and is planned to make the move from 2nd generation systems smoother, yet in the end, supplant them. This implies UMTS will, in the long haul, bolster all applications at present served by 2nd generation cell frameworks, for example, GSM and PDC, cordless frameworks like DECT, and satellite frameworks like IRIDIUM (WCDMA for UMTS, 2004).

The UMTS Network comprises Radio Network Controllers (RNC), Radio Base Stations (Node B), an O&M system including RANOS (Radio Access Network Operation System), TRAM (Tools for Radio Access Management) and Core Network (CN) as shown in Figure 1.1.

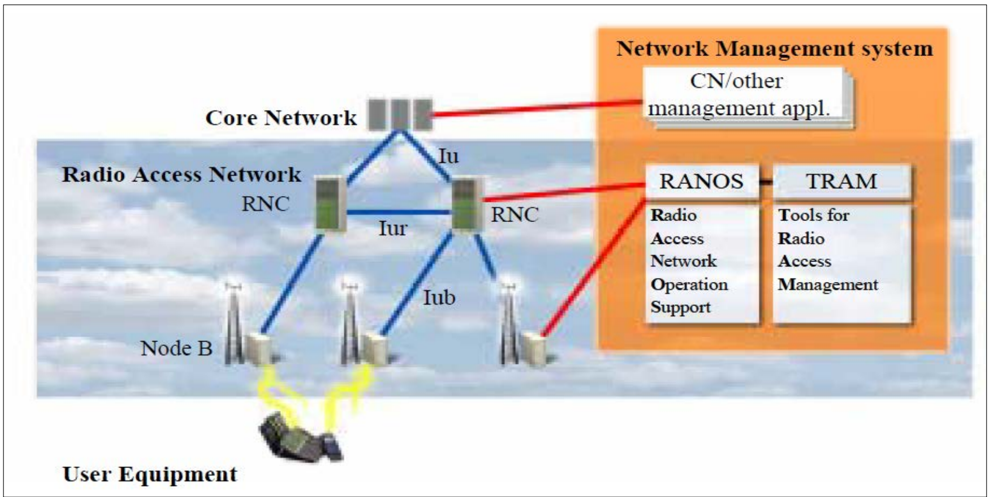

Figure 1.1 UMTS Network (Ericsson WCDMA/UMTS System Overview, 2001)

Wideband code division multiple access (WCDMA) radio access technology used in UMTS offers greater spectral efficiency and bandwidth. The frequency from 400 MHz to 3 GHz is allocated for communication spectrum. The following frequency bands are currently identified for UMTS in all three ITU Regions: 450 – 470 MHz, 790 – 960 MHz, 1710 – 2025 MHz, 2110 – 2200 MHz, 2300 – 2400 MHz and 2500 – 2690 MHz. Additional frequency bands identified for IMT on a Regional or National basis are 698-790 MHz (Region 2), 610 – 790 MHz (9 countries in Region 3: Bangladesh, China, Rep. of Korea, India, Japan, New Zealand, Papua New Guinea, Philippines and Singapore), 3400 – 3600 MHz (Over 80 Administrations in Region 1 plus 9 in Region 3 including India, China, Japan and Rep. of Korea). (WCDMA for UMTS, 2004).

In the direction of these developments, data transfer for downlink connection is 384 Kbps for the first release in 1999 (R99, the first UMTS release), and 7.2 Mbps for future release in which HSDPA technology, also called 3.5G, was introduced. For the uplink, the data transfer ranges from 384 Kbps to 5.76 Mbps theoretically (3GPP, 1999). The latest release (Release 11) of UMTS can provide peak data rates up to 337.5 Mbps in the downlink and 22 Mbps in the uplink, using a combination of air interface improvements as well as multiple cell and Multiple Input Multiple Output features together (3GPP, 2014).

The new technology provides greater bandwidth and higher throughput. Therefore, more capacity is required to meet higher user expectation. It is also obvious that crowded games, special events, and high-tech conventions are notorious for overwhelming cell-phone networks with up and down of traffic. In such cases, linear planning of network and classical prediction methods do not work well and the design of the networks come under surge of unusual things.

So, networks need to be well designed, dimensioned and deployed to cope with expected or unexpected situations. Most important necessities for these tasks are developing accurate traffic models and making reliable traffic prediction. More accurate traffic prediction means that more realistic results can be achieved.

Predictive modelers also estimate future traffic demands and help telecommunications carrier plan accordingly. They start by collecting a carrier's data to understand what has gone on in a network and what it looks like now—how much traffic is transmitted, what percentage is voice or video or text, what path it takes through the network. Then they run simulations to assess the impact if, for instance, a carrier starts selling the iPhone, or changes its marketing plan, or moves from 3G to 4G services.

1.2. Literature

1.2.1 Previous Work on Network Traffic Prediction

In (Chan, et. al., 2006), an access control protocol has been proposed for an integrated voice, video, and non-real-time data traffic on the forward link (cell-site to mobile). The protocol contains predicting the remaining capacity available for the HSDPA packet data traffic. It evaluates the traffic models based on Markovian, AR and TSMR processes. Among the three models, the AR and TSMR show higher performance compared with the Markov model.

In (Buerger et. al., 2008), four different models including linear, exponential, ARMA and DHR methods have been used for forecasting packet services traffic in 3G networks. It has been shown that sophisticated models (i.e. models based on ARMA and DHR) deliver better results than the simple ones (i.e. models based on linear and exponential functions).

In (Gowrishankar, 2009), short time network traffic prediction has been studied. RRBFN, ESN and FARIMA methods have been applied for developing prediction models. The results show that neural network predictors show better performance than statistical models.

In (Chen et. al., 2010), modeling and forecasting have been accomplished by using BP neural network. The model has been built by analyzing the characteristics of network traffic and the results show that the model has good astringency and stability.

In (Syed et. al., 2010), wavelet filters based on multi-resolution analysis along with the Seasonal Autoregressive Moving Average (SARIMA) models have been used to forecast traffic. The results have been compared with the ones obtained by simple SARIMA and it is concluded that the proposed methodology gives more accurate forecast.

In (Chabaa et. al., 2010), to analyze internet traffic, multi-layer perceptron (MLP) based model has been developed. For estimating the weights of the neurons within the model, Levenberg-Marquardt (LM) and the Resilient back propagation (Rp) algorithms have been applied and the efficiency of these models have been compared with the one of some other statistical models.

In (Tan et. al., 2012), bittorrent type network traffic and its behavior have been investigated by using time series ARMA model. It is proved that bittorrent network traffic can be modeled and forecasted by ARMA model effectively.

In (Kim et. al., 2011), an autoregressive-generalized autoregressive conditional heteroscedasticity (AR-GARCH) error model for forecasting internet traffic has been developed and its performance has been compared with seasonal autoregressive integrated moving average (ARIMA) models in terms of root mean square error (RMSE) criterion. The results indicated that the seasonal AR-GARCH models outperformed the seasonal ARIMA models in terms of forecasting accuracy with respect to the RMSE criterion.

In (Chen et. al., 2012), the flexible neural tree (FNT) has been utilized to predict short time scale traffic in the network. FNT structure has been developed by using genetic programming (GP) and internal model parameters have been optimized by Particle Swarm Optimization (PSO) algorithm. It is concluded that the developed model is efficient and reliable for short time scale traffic measurement.

In (Miguel et. al., 2012), long term internet traffic has been predicted by using ensembles of Artificial Neural Network. Four different TLFNs have been utilized and each differ from others by the training data and the number of artificial neural networks used in the forecast. Results obtained by the TLFNs models have been also compared with the ones obtained with the classical Holt-Winters method. The proposed neural network models perform well and can be a good option for the link that transports internet traffic.

In (Cortez et. al., 2012), the authors have utilized Naïve-Benchmark, Holt-Winters, ARIMA, and ANN to figure Internet activity. It has been inferred that while ANN-based models gave the best outcome for 5-minute and hourly sets, Holt-Winters based models were more exact than the ones obtained by the other strategy for the daily set.

In (Oliveira et. al., 2014), MLP and SAE have been used to forecast Internet traffic. The results of MLP and SAE based models have been confronted. The outcomes have demonstrated that despite the fact that SAE was a complex method, MLP based models performed well with respect to the SAE.

In (Katris et. al., 2015), several different approaches have been evaluated for internet traffic forecasting. Firstly, the dependence of short or long and non-linearity has been explored to take the advantage of such information. FARIMA, MLP, RBF, Holt-Winters, ARIMA/GARCH, FARIMA/GARCH, hybrid FARIMA+RBF, and hybrid FARIMA+MLP are the models used in the study. The results show that forecasting models that use non-linear functions has a better performance for internet traffic prediction.

The summary of the studies and the methods used in each study are given in Table 1.1.

Table 1.1 Summary of studies between the years 2006 and 2016

Study	Methods	Traffic Type
(Chan, *et.al.*, 2006)	Markovian, AR, TSMR	UMTS voice and data
(Buerger *et.al.*, 2008)	ARMA, DHR	UMTS data
(Gowrishankar, 2009)	RRBFN, ESN, FARIMA	UMTS data
(Chen *et.al.*, 2010)	BP-ANN	Internet, IP based data
(Syed *et. al.*, 2010)	Wavelet Filter based SARIMA, SARIMA	Intranet, IP based data
(Chabaa *et. al.*, 2010)	MLP, LM, RBP	Internet, IP based data
(Tan *et. al.*, 2012)	ARMA	Internet, IP based data
(Kim, 2011)	AR-GARCH, ARIMA	Internet, IP based data
(Chen, 2011)	GA, GNN	Internet, IP based data
(Chen *et. al.*, 2012)	FNT, FFNN	Internet, IP based data
(Miguel *et. al.*, 2012)	ANN, Holt-Winters, TLFN	Internet, IP based data
(Cortez *et. al.*, 2012)	Naive Benchmark, ANN, MLP, Holt-Winters	Internet, IP based data
(Maurya et. al., 2012)	FIS	Internet, IP based data
(Wang *et. al.*, 2012)	NTWD, ARMA without Wavelet Analysis	Internet, IP based data
(Dorgbefu jnr. *et. al.*, 2013)	Kalman filter	UMTS data
(Oliveira *et. al.*, 2014)	MLP, SAE	Internet, IP based data
(Kamińska-Chuchmała, 2014)	GE	Internet, IP based data
(Katris *et. al.*, 2015)	FARIMA, MLP, RBF, ARIMA/GARCH, Holt-Winters	Internet, IP based data

GLLA, Generic Local Linear Approximation; **RBF**, Radial Basis Function Neural Network; **SVM**, Support Vector Machine; **LLA**, Local Linear Approximation; **ARMA**, Auto-Regressive Moving Average; **MMPP**, Markov-Modulated Poisson Process; **FFNN**, Feed-Forward Neural Network; **WMRA**, Wavelet Multiresolution Analysis; **ARIMA**, Autoregressive Integrated Moving Average; **FIR,** Multiresolution Finite-Impulse-Response; **MODWT,** Maximal Overlap Discrete Wavelet Transform; **MMLP**, Multiresolution Multilayer Perceptron; **ANN**, Artificial Neural Network; **MLP**, Multilayer Perceptron; **GA-RBF**, Genetic Algorithm RBF; **BP**, Back Propagation; **LS- SVM**, Least Square SVM; **FARIMA**, Autoregressive Fractionally Integrated Moving Average; **SPN**, Spinning Network; **GM(1,1)**, Grey-Markow Model; **ANFIS**, Adaptive Neuro-Fuzzy Inference System; **GARCH**, Generalized Autoregressive Conditional Heteroscedasticity; **SVR**, Support Vector Regression; **GML**, Gaussian Maximum Likelihood; **ES**, Holt Exponential Smoothing; **MA**, Moving Average; **AR**, Auto-Regressive

1.3. Motivation, Purpose and Contributions of This Thesis

Traditionally, prediction of network-based traffic which is collected from different sources is often performed by statistical methods, such as linear and exponential regression. However, these methods have limited efficiency. In other words, forecasts based on these models have a strong deviation compared with actual data collected from the live network. Therefore, machine learning methods including supervised learning models are appropriate solutions and they can give more sophisticated and reliable estimations with respect to the classical methods.

In literature, there exist several studies which predict the network traffic with the help of statistical as well as machine learning regression methods, but there is no comprehensive work that covers both voice (circuit switched) and data (packet switched) traffic together within the same framework.

The aim of this thesis is to derive models for traffic characteristics of a 3G network which is commercially deployed in Turkey and predict voice and data traffic by using various machine learning methods. The machine learning methods that have been employed are Support Vector Machines (SVM), Multilayer Perceptron (MLP), Random Forest (RF) and Radial Basis Function Neural Network (RBF). Additionally, the Holt-Winters method has been applied to develop prediction models. Finally, the performance of statistical and machine learning regression methods have been compared in this thesis.

The scope of this thesis is differentiated from the studies in literature as outlined below:

- To the best of our knowledge, this is the first study ever that has developed traffic prediction models for one of the mobile operators in Turkey using machine learning methods.
- While UMTS network traffic prediction was also employed in the previous studies, this is the first study that ever used machine learning methods such as SVM, MLP, RBF, RF with a statistical method Holt-Winters and their prediction results have been compared using a performance indicator known as *MAPE.*
- To the best of our knowledge, this is the first research study that predicts both voice traffic and packet data traffic together within the same framework for a 3G mobile network.

- To the best of our knowledge, short, mid and long term three different time granularity traffic prediction methods have been focused on this study while previous works do not cover the time granularity as wide as this study.

2. DATASET GENERATION

In this thesis, three different types of the dataset have been used to build traffic prediction models of a commercially deployed 3G network operating in Turkey. Each dataset contains different time scales including hourly, daily and weekly traffic values. The sets and their traffic contents have been acquired from the live 3G mobile operator network with the permission of service provider.

1st data set: The first dataset includes hourly basis data for both voice and packet data (IP based) traffic acquired from a live network. The voice set contains circuit-switched (CS) traffic volume in Erlang, while packet-switched (PS) set contains IP-based data traffic volume in bits. PS datasets contain three subsets, which are PS Downlink, PS Uplink and the sum of these, called PS Total traffic. The dataset was saved every hour between October 24th, 2015 and January 19th, 2016.

2nd data set: The second dataset includes voice and packet switched traffic with daily acquired values. It consists of CS (Voice) traffic, PS Downlink, PS Uplink and PS Total daily network traffic, respectively. It was acquired from the 3G Network between December 21st, 2014 and May 8th, 2016.

3rd data set: The third dataset consists of voice and packet switched traffic including CS traffic, PS Downlink, PS Uplink and PS Total traffic with weekly acquired values. It contains weekly traffic between March 3rd, 2014 and May 2nd, 2016.

The distributions of traffic for different datasets are given in Figure 2.1 through Figure 2.12, respectively.

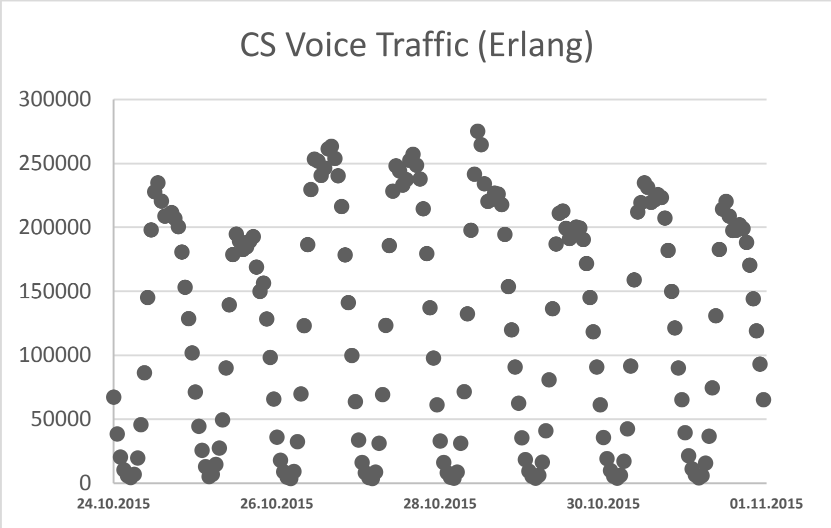

Figure 2.1 Hourly CS voice traffic distribution

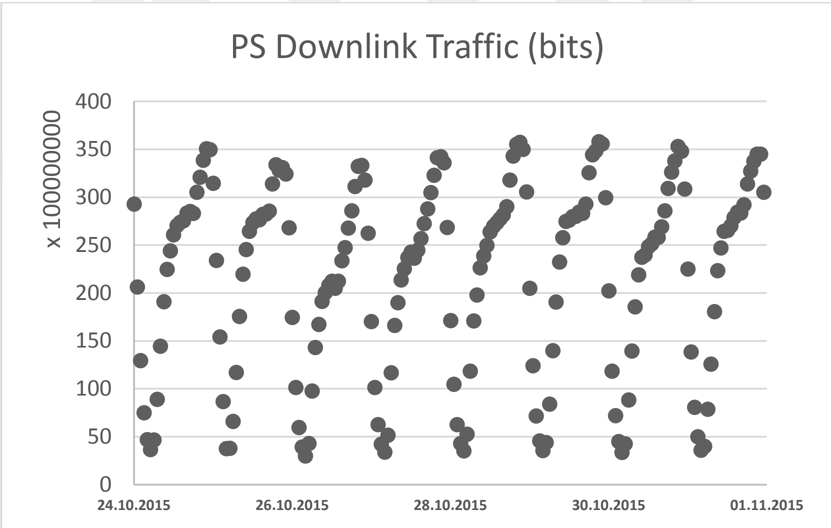

Figure 2.2 Hourly PS downlink data traffic distribution

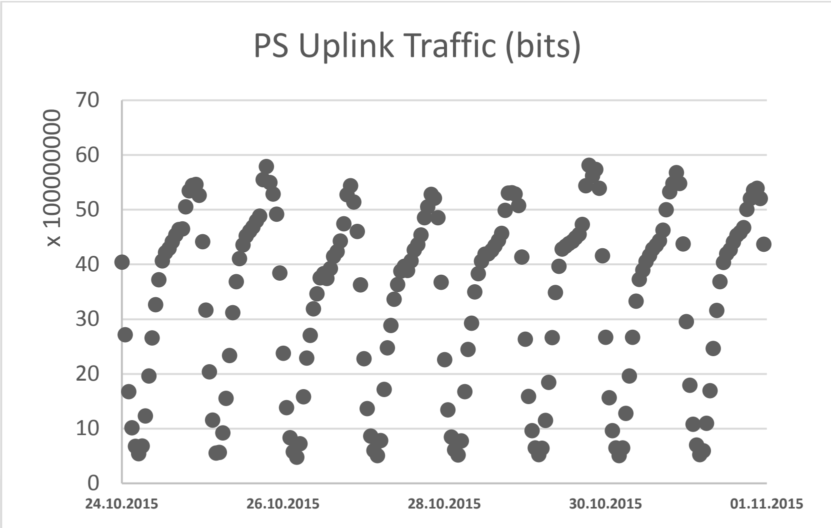

Figure 2.3 Hourly PS uplink data traffic distribution

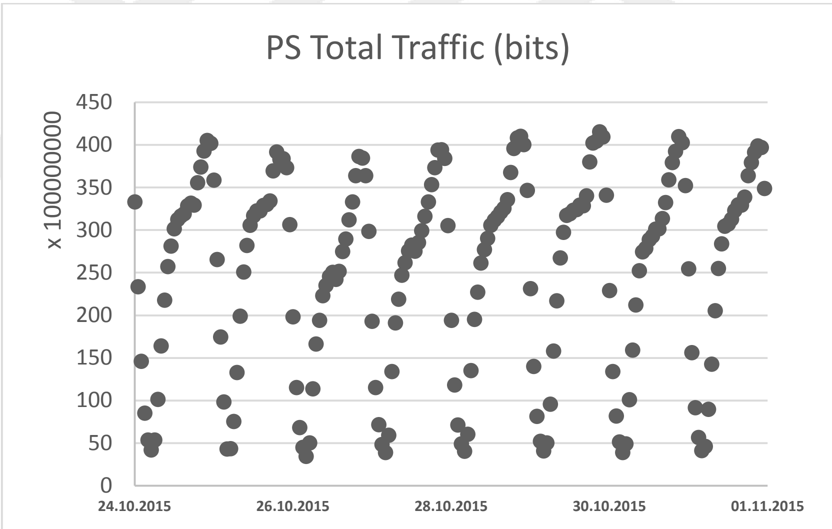

Figure 2.4 Hourly PS total data traffic distribution

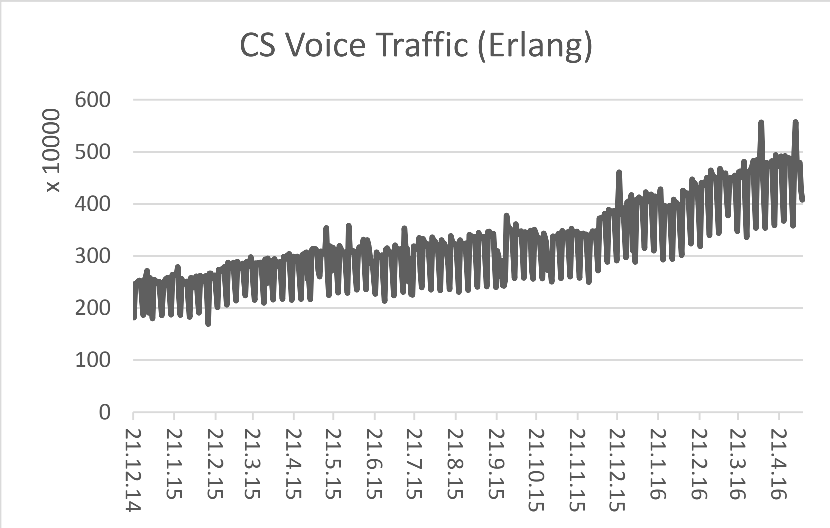

Figure 2.5 Daily CS voice traffic distribution

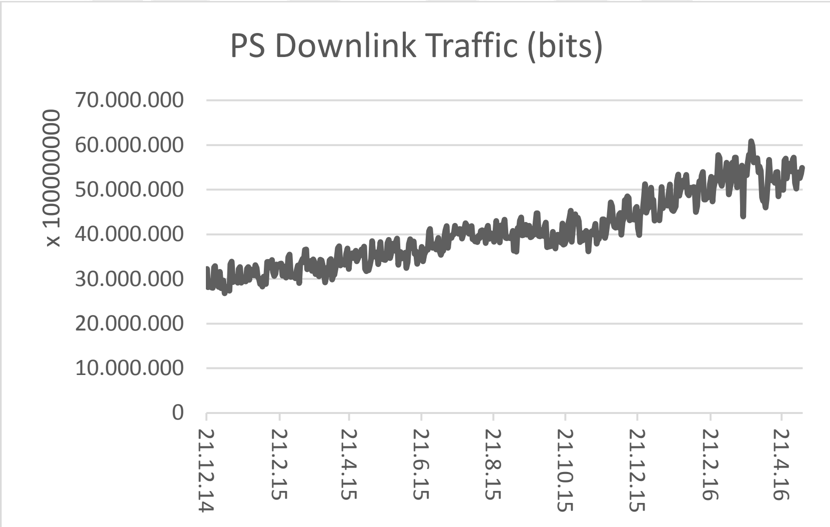

Figure 2.6 Daily PS downlink data traffic distribution

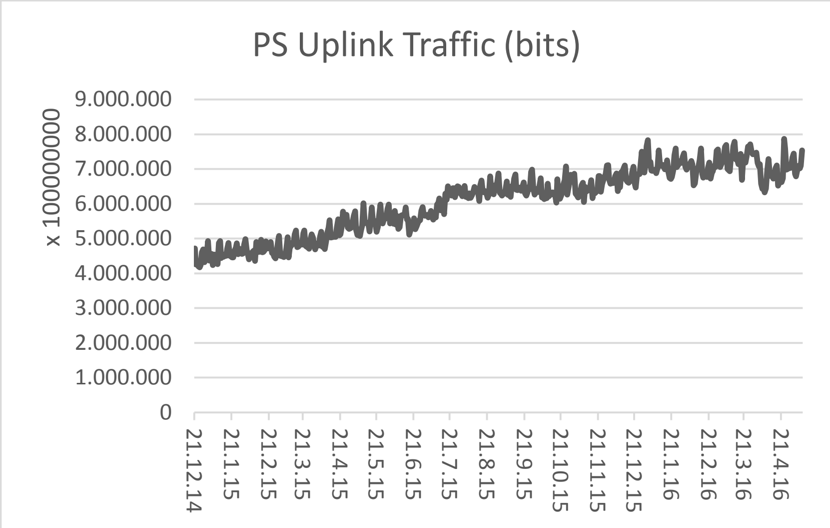

Figure 2.7 Daily PS uplink data traffic distribution

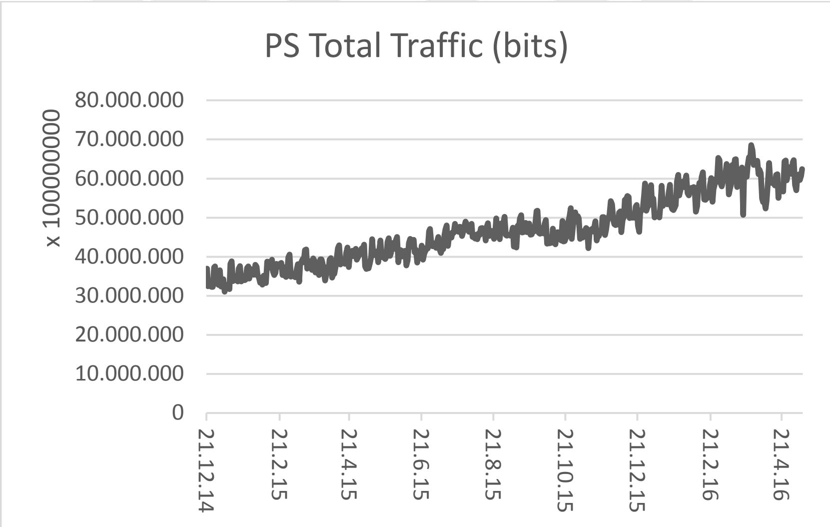

Figure 2.8 Daily PS total data traffic distribution

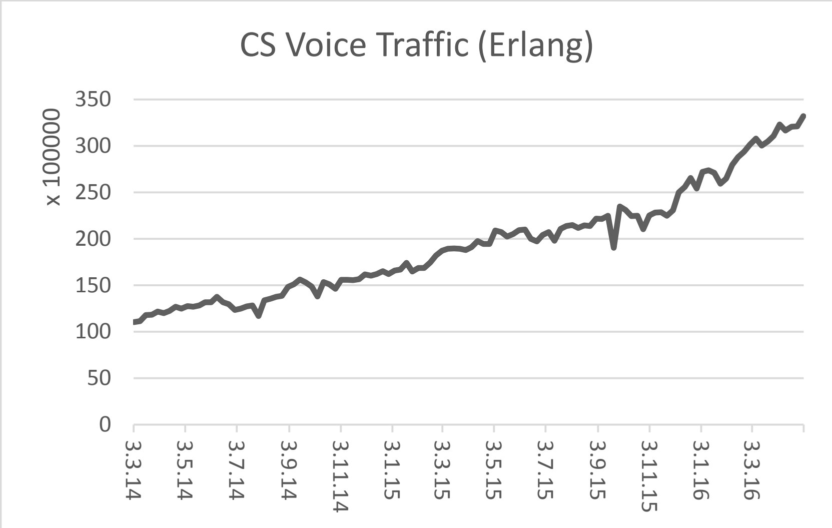

Figure 2.9 Weekly CS voice traffic distribution

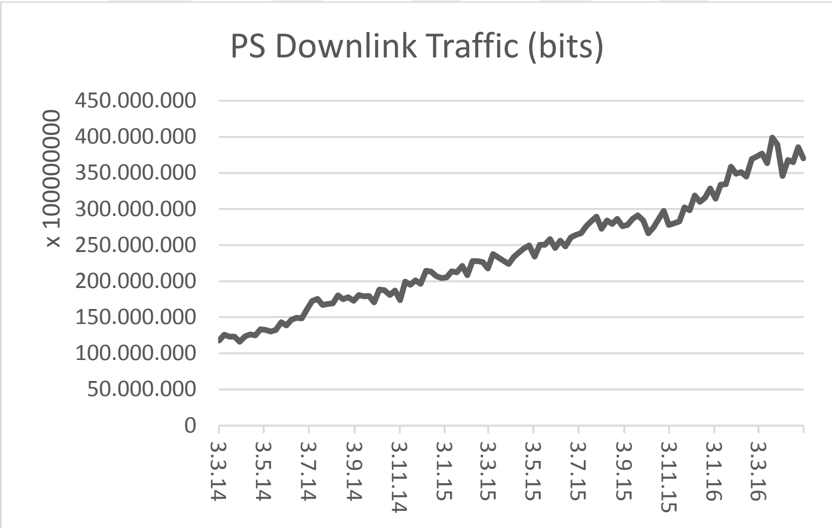

Figure 2.10 Weekly PS downlink data traffic distribution

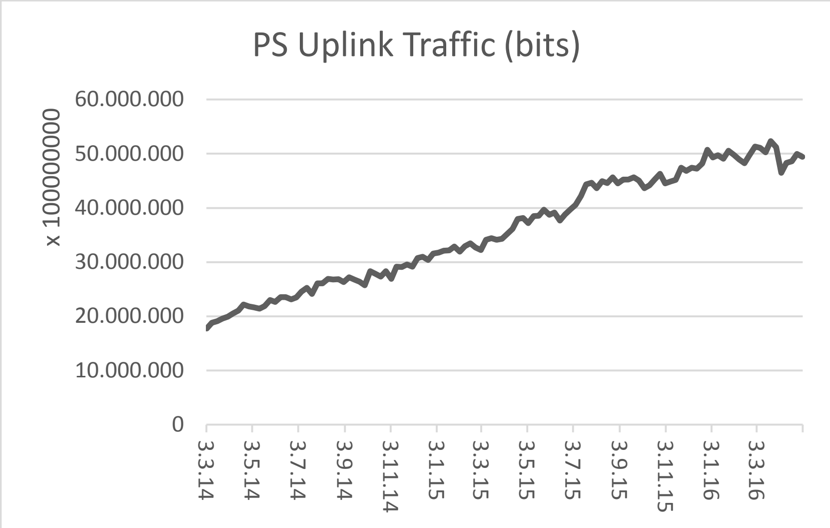

Figure 2.11 Weekly PS uplink data traffic distribution

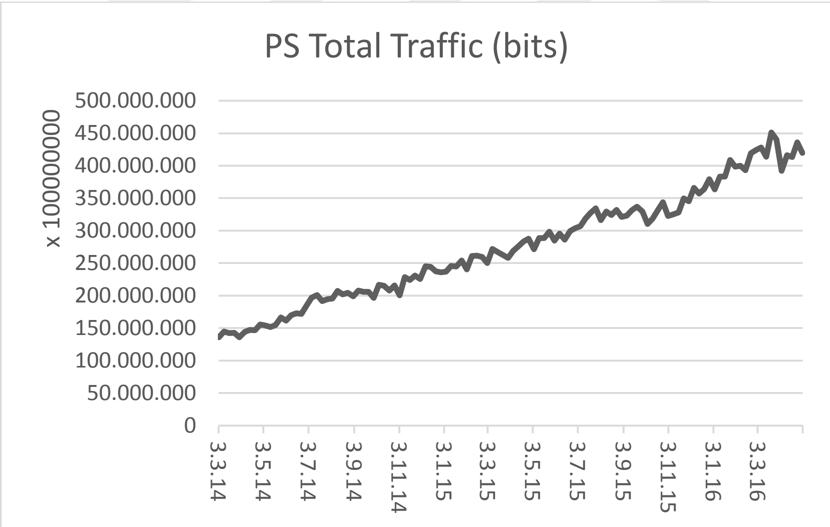

Figure 2.12 Weekly PS total data traffic distribution

3. OVERVIEW OF METHODS

3.1. Support Vector Machine

SVMs were introduced by Vladimir N. Vapnik and Alexey Ya. Chervonenkis in 1963 and are supervised learning algorithms for both the solution of classification and regression problems.

3.1.1 Linear SVM

Assume (x_1, y_1) ... (x_n, y_n) represent the training dataset where x_i are the vectors for the observations and y_i = {-1, +1} be the targets. The primary goal of an SVM is to find the optimal hyperplane (i.e. hyperplane with the highest margin) that separates the two distinct targets from each other. The margin of the hyperplane should be selected in such a way that distinct targets are as far as possible from each other. The idea of choosing a large margin for the hyperplane is to provide a more resistant hyperplane to noise.

Assume that all the data satisfy the constraints given by (3.1.) and (3.2.)

$$w.x_i + b \geq +1 \quad y_i = +1 \tag{3.1.}$$

$$w.x_i + b \leq +1 \quad y_i = -1 \tag{3.2.}$$

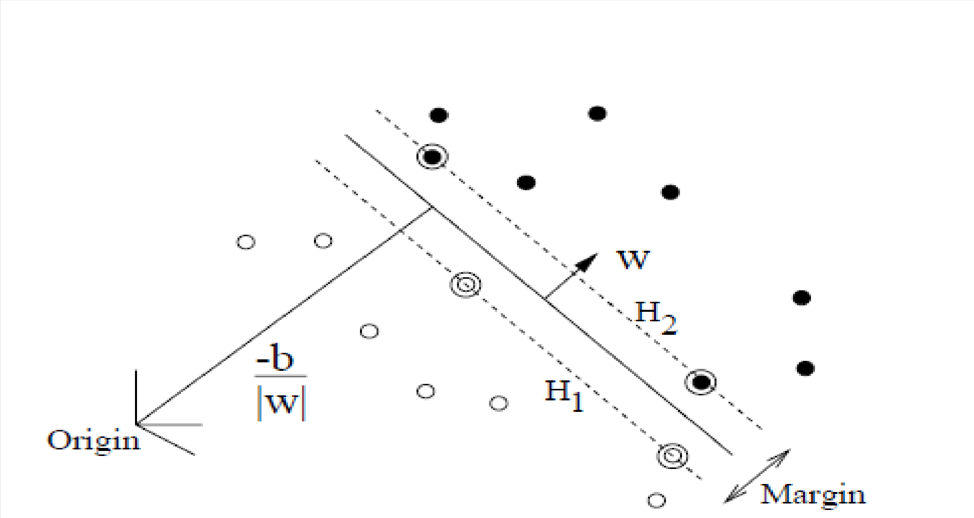

Figure 3.1 Linear separable hyperplane. Support vectors are circled.

(3.3.) shows the combination of the two constraints,

$$y_i\ (w.x_i + b) \geq\ 1 \quad \forall_i \tag{3.3.}$$

In (3.3), the normal to the hyperplane is given by w, the perpendecilur distance from the origin is given by $|b|\ /\ ||w||$, and the Euclidean form of w is given by $||w||$ is. The margin, which the distance between H_1 and H_2 and given by p, can be calculated with (3.4.):

$$\rho = \frac{|1-b|}{||w||} - \frac{|-1-b|}{||w||} = \frac{2}{||w||} \tag{3.4.}$$

The optimum value of the margin can be found by solving the primal optimization problem given in (3.5.)

$$\min_{w \in \mathcal{H}} \tau(w) = \frac{1}{2}\left||w^2|\right|$$

subject to

$$y_i\,(w.x_i + b) \geq 1 \quad \forall_i \tag{3.5.}$$

Then Lagrangian technique is used to solve (3.5.) by introducing new Lagrange multiplier α_i for each constraint and the new formulation of the minimization problem is given by,

$$\min_{w,b} L(w,b,\alpha) = \frac{1}{2}\left||w|\right|^2 - \sum_{i=1}^{l} \alpha_i\, y_i(x_i w + b) + \sum_{i=1}^{l} \alpha_i \tag{3.6.}$$

with $\alpha_i \geq 0$ for each constraint in (3.5.). The problem is then reduced to minimizing (3.6.) with respect to *w*, *b* and at the same time requiring the derivatives of $L(w,b,\alpha)$ with respect to all the α vanish.

The solutions w^*, b^* and α^* should satisfy the conditions given in (3.7.) through (3.9.) with respect to the Karush-Kuhn-Tucker (KKT) conditions:

$$\frac{\partial L(\boldsymbol{w}^*, b^*, \alpha^*)}{\partial \boldsymbol{w}} = w_v - \sum_i \alpha_i\, y_i x_{iv} = 0 \;\; for\ v = 1, \ldots, d \tag{3.7.}$$

$$\frac{\partial L(\boldsymbol{w}^*, b^*, \alpha^*)}{\partial b} = -\sum_i \alpha_i\, y_i = 0 \tag{3.8.}$$

$$y_i(\boldsymbol{x}_i.\boldsymbol{w} + b) - 1 \ \geq \ 0, \quad \forall_{\mathrm{i}}$$

(3.9.)

$$\alpha_i \ \geq \ 0 \quad \forall_i$$

(3.10.)

$$\alpha_i(y_i(\boldsymbol{x}_i.\boldsymbol{w} + b) - 1) = \ 0 \quad \forall_i$$

(3.11.)

It is to be noted that the given KKT conditions are enough for w^*; b^*; α^* to be a solution. Therefore, if one can find a solution to the KKT conditions, that solution is also valid for the SVM problem. The optimal hyperplane as a linear combination of the vectors in the training set is defined by the first KKT condition as

$$\boldsymbol{w}^* = \sum_i \alpha_i y_i \, \boldsymbol{x_i}$$

(3.12.)

The second KKT condition requires that the α_i coefficients of the training instances should satisfy

$$\sum_{i=1}^{n} \alpha_i^* y_i = 0 \,, \ \ \alpha_i \ \ \geq 0$$

(3.13.)

Maximizing (3.6.) according to α, and minimizing with respect to w and b is a dual solution of the SVM problem. The dual formula can be obtained by placing (3.7.) and (3.8.) into (3.6.)

$$\operatorname*{Max}_{\alpha} L_D = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \boldsymbol{x_i}.\boldsymbol{x_j} \quad subject\ to \ \ \forall_i \begin{cases} \sum_i \alpha_i y_i = 0 \\ \alpha_i \geq 0 \end{cases}$$

(3.14.)

Thus, by solving the equations, the coefficients α_i is obtained. The conditions with $\alpha_i > 0$ are called “support vectors”. This leads to the decision function

$$f(x) = w^T x_i + b$$

$$= \sum_{i=1}^{M} y_i \, \alpha_i \left(x_i^T x \right) + b$$

(3.15.)

3.1.2 Non-Linear SVM

A linear separating hyperplane cannot usually divide most of the datasets. However, these datasets can be linearly divided if they are mapped into a higher dimensional space by a mapping function Φ(x) and building a separating hyperplane with a maximum margin in the input space. Figure 3.2 shows the relation between linear decision function in the future space and non-linear decision boundary in the input space.

Several Kernel functions including linear kernel, polynomial kernel, RBF kernel and sigmoid kernel can be used to calculate the hyperplane given in (3.16.).

$$K(x, y) = \langle \Phi(x), \Phi(y) \rangle \qquad (3.16.)$$

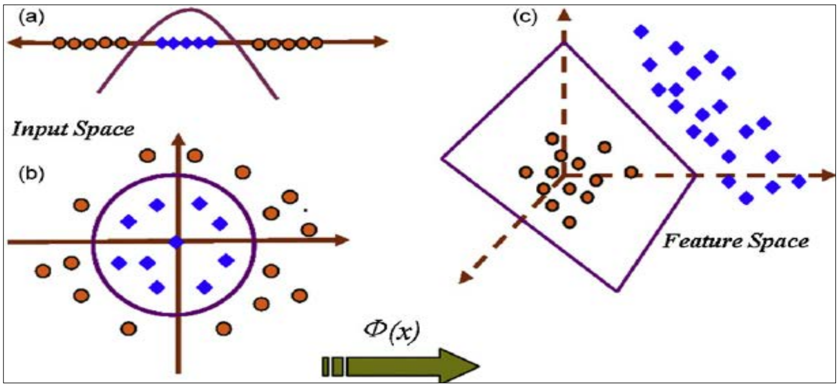

Figure 3.2 Data not linearly separable in the input space (a), separable by non-linear space (b), Kernel mapped feature space (c).

3.2. Multi-Layer Perceptron

Multi-layer perceptrons are organized in layers of neurons and implement a feed-forward processing chain. Following notations can be used for the layers and nodes of an MLP:

- The network consists of L layers, with l=0 denoting the input layer and l=L denoting the output layer.
- The notation for a single node is n_i^l $(1 \leq i \leq N^l)$, N^l being the number of nodes in layer l.
- The activation of a network node depends on the strength of the input to that node with respect to threshold value. For notational convenience, the network thresholds are treated uniformly by adding an extra node with a fixed output of 1.0 to all but the output layer. This node – called the *bias unit* – is denoted n_0^l (for $\neq L$).
- To allow for MLPs of arbitrary connectivity, it is useful to define a set of *source nodes* S_i^l and set of *target nodes* T_i^l for each node n_i^l. Given that node $m_j^m$is connected node n_i^l, n_j^m is a source node of n_i^l(i.e. $n_j^m \in$

S_i^l) if $m<l$, but a target node of n_i^l (i.e. $n_j^m \in T_i^l$) if $m>l$. Set S_i^l is null for all input nodes (i.e. $S_i^0 = \emptyset$ for i =1,…,N^0) and for all bias units (i.e. $S_i^0 = \emptyset$ for l =0,…,L-1); set $T_i^l$is null for all output nodes (i.e. $T_i^l = \emptyset$ for i =1,…,N^L).

- Network weights can be represented in terms of the nodes they connect; thus weight w_{ij}^{lm}connects nodes $n_j^m$and n_i^l with $m<l$ (i.e. $n_j^m$is a source node of $n_i^l$and $n_i^l$is a target node of n_j^m). However, it will often be more convenient to consider weights in terms of the weight vector **w** comprising all W weights in the network, with a single weight denoted $w_i (1 \le i \le W)$.

The number of nodes in the input and output layers of the MLP is determined by, respectively, the pattern size and the target size of the chosen training task. MLPs are typically trained using fixed training set of P training pairs, with each training pair comprising two real valued vectors – a pattern $p_q (1 \le q \le P)$ and corresponding target (desired output) t_q. Individual pattern and target elements are denoted $p_{i,q}$ $(1 \le i \le N^0)$ and $t_{j,q}$ $(1 \le j \le N^L)$ respectively. The output $y_{i,q}^0$ of input node i is simply $p_{i,q}$ for pattern q (except for y_0^0 the fixed output of the bias unit). For non-input node n_i^l, the output is given by the weighted sum

$$a_{i,q}^l = \sum_{n_j^m \in S_i^l} w_{ij}^{lm} y_{j,q}^m, \qquad l > 0 \tag{3.17.}$$

$$y_{i,q}^l = f\left(a_{i,q}^l\right), \qquad l > 0 \tag{3.18.}$$

where $a_{i,q}^{l}$ is the activation of node n_{i}^{l} for pattern *q*, and the *squashing* or *activation function f(x)* is both monotonic (i.e. non-decreasing) and continuously differentiable. By far the most commonly used squashing function is the *sigmoid* or *logistic function*

$$f(x) = \frac{1}{(1 + e^{-x})} \tag{3.19.}$$

which compresses the output of each non-input node in the range [0,1]. The most popular alternative to the sigmoid is the *hyperbolic tangent*, *f(x)* = tanh(x), which gives a compressed range [-1, 1].

The layers between the input and output layers are known as *hidden layers*. The number of hidden layers and nodes has a major impact on MLP training: too few, and the network will be unable to learn the problem; too many, and the network may take excessively long to train and have poor *generalization* capabilities – a features as, but are not identical to, patterns in the training set. Upper and lower bounds on the number of hidden nodes required for an MLP to be capable of learning a given task have been established by Huang and Huang (1991), but optimal number of hidden nodes is much more difficult to determine.

3.3. Radial Basis Function Network

3.3.1 Network Architecture

RBF network includes several layers and the first layer has input neurons. These neurons feed the feature vectors into the network. The second one is the hidden layer that calculates the result of the basic functions. The last layer is the output layer that calculates a linear union of the basic functions. These kinds of networks have the general estimation property (Park and Sandberg, 1991). Simple structures of these networks give a decreasing for training time and make possible learning in stages.

$X \in R^n$ (a vector) is an input and $f(x)$: $R^n \rightarrow R$ (a function) is an output estimated by using an RBF network:

$$f(x) = \sum_{i=1}^{N} w_i h_i\,(x) \tag{3.20.}$$

where N represents the number of neurons and $h(x)$ is the radial basis function in the hidden layer. The typical radial basis function is taken to be Gaussian:

$$h_i(x) = exp\left(-\frac{(x-c)^2}{r^2}\right) \tag{3.21.}$$

The parameter c is the center vector and r is its radius.

3.3.2 Training

A two-step algorithm is usually used to train RBF. The first step requires the selection of the center vectors $\boldsymbol{c_i}$ of the RBF functions in the hidden layer. Randomly sampled from some set of examples or K-means clustering can be used to accomplish the first step.

In the second step, the coefficients $\boldsymbol{w_i}$ are fitted to the hidden layers output with respect to some transfer function. The least squares function is one of the commonly used transfer functions. Detailed information can be found in “Multivariable Functional Interpolation and Adaptive Networks”, (Broomhead D. S., 1988).

3.4. Random Forest

RF was built by Leo Breiman (2001). A group of unpruned classification or regression trees which are formed by the random choice of samples of the training data is created with the RF (Ali et. al., 2012).

RF makes use of bagging and random feature selection, which are two important machine learning methods. In bagging, every tree is practiced on the training set's bootstrap case and estimations are generated by a large number of trees.

RF is a method that is an extension of bagging. While growing a tree, a subset of features is selected randomly by RF instead of using all features to divide into each node. To assess the expectation execution of the RF calculation, RF utilizes a kind of cross-approval correspondingly to preparing process by using OOB cases. Particularly, at the training, a certain bootstrap sample has been used during growth of each tree. Since bootstrapping is exemplifying by replacing with the training set, a certain set of the sequence is repeated in the case when others are "left out" of the samples. The "left out" ranks generate the OOB case. In the mean, while each tree is grown, $1- e^{-1} \cong 2/3$ of the training sequences has been used and $e^{-1} \cong 1/3$ has been left as OOB. Since OOB ranks haven't been employed during construction of the tree, they can be employed to assess the estimation performance (Jiang et. al., 2007).

RF algorithm is shown below;

1. Choose ${}_{tree}m$ bootstrap cases by using the main data.
2. For every case, raise an *unpruned* regression or classification tree by regulating sequent change: in every node, instead of selecting the best division between all predictors, say ${}_{try}n$ of the predictors, select the best division through those attributes.

3. Guess new data by collecting predictions of the m_{tree} trees (that is, if it is a classification problem, then choose majority, if it is a regression problem, then choose average).

The estimated error rate could be attained, according to the training set as below;

1. At every bootstrap repetition, guess the data in the OOB using the tree risen with the bootstrap sample.
2. Collect the OOB estimations. Figure out the error rate, and define it as the OOB estimate ones (Liaw and Wiener, 2002).

3.5. Holt-Winters

Holt-Winters is an exponential smoothing method which is used when the data exhibits both trend and seasonality. Seasonal and trended samples are separated from the unnecessary attributes by meaning the historical rates. It has some benefits like the easiness of using it, having less computation, and more accurate results for seasonal series.

The two main HW models are *Additive model* for time series exhibiting additive seasonality and *Multiplicative model* for time series exhibiting Multiplicative seasonality.

The general forecast function is:

$$\hat{y}_{t+l|t} = (m_t + l b_t) c_{t-s+l} \qquad l = 1,2 \ldots \tag{3.22.}$$

$$m_t = \alpha_0 \frac{y_t}{c_{t-s}} + (1 - \alpha_0)(m_{t-1} + b_{t-1}) \tag{3.23.}$$

$$b_t = \alpha_1 (m_t - m_{t-1}) + (1 - \alpha_1) b_{t-1} \tag{3.24.}$$

$$c_t = \alpha_2 \frac{y_t}{m_t} + (1 - \alpha_2) c_{t-s}$$

(3.25.)

where m is the component of level, b is the component of the slope, c is the relevant seasonal component with s signifying the seasonal period (e.g. 4 for quaerterly data and 12 for monthly data), $\alpha_0, \alpha_1, \alpha_2$ are model parameters lie between 0 and 1.

It is important to select the starting values and smoothing parameters (Chatfield and Yar (1998). For starting values, the component m_0 is sensible to the avearege observations in the first year, i.e.

$$m_0 = \sum_{t=1}^{s} \frac{y_t}{s}$$

(3.26.)

where the number of seasons is given by s. 3.27. can be used to find the starting value for the slope

$$b_0 = \frac{\{\sum_{t=1}^{s} y_t/s\} - \{\sum_{t=s+1}^{2s} y_t/s\}}{s}$$

(3.27.)

After allowing a trend adjustment, the seasonal index rate can be computed:

$$c_0 = \frac{\{y_k - (k-1) b_0/2\}}{m_0}$$

(Multiplicative)

$$c_0 = y_k - \left\{ m_0 + \frac{(k-1)b_0}{2} \right\}$$

(Additive)

where k=1, 2…, s and it'll lead to s separate values for c_0 which is what is required to gain the initial seasonal pattern.

Usually, the values between 0.02 and 0.2 are used for the smoothing parameters. It is again possible to estimate them by minimizing the sum of squared one-step-ahead errors, but there is no exclusive combination of $\alpha_0, \alpha_1, \alpha_2$ which will minimize the square errors for all t.

4. DEVELOPMENT OF PREDICTION MODELS

4.1. SVM Model

Support Vector Machines are learning machines that depend on two important elements: a linear learning algorithm and specific kernel that calculates the internal process of input data points and projects them to a future space. For the learning algorithm, the penalty parameter “C” controls the trade-off between margin maximization and error minimization (Chapelle et.al. 2002)

The selection of kernel type and the parameter C are the critical steps for the learning mechanism. The accuracy of the whole process also depends on these tasks. There is a tradeoff between reducing the error related to training and reducing the complexity of the model and this is stated by parameter C. A small value of C will increase a number of training errors and a large value of C will exhibit an attitude similar to that of a hard-margin SVM. The kernel parameters are important in the sense that they act as a bridge between the input space and the high-dimensional feature space (Ji and Wang, 2007). “ε” is the another important parameter for the SVM. It controls the width of insensitive zone and states the number of support vectors.

In this thesis, RBF and polynomial kernel have been utilized separately for developing the models. Recently, polynomial kernels are less widely used than the RBF kernel. The reason is that in the case of training, a polynomial kernel may have less accuracy than that of RBF, but training with a low degree polynomial kernel strategy is much faster and it saves time.

These all parameters together with the penalty parameter “C” are called the hyperparameters of the SVM.

The optimal values hyperparameters are usually found by grid search. That is, the values of the parameters are differentiated with a fixed step size through an interval of values and the performance of every combination is assessed using some performance measure. A cross-validation within the grid search can be utilized in order to develop the generalization capability of the SVM regression model. In the process of k-fold cross validation, the cross validation is recurred k times (this means k folds), with each of the k subsets utilized precisely once as the validation data. The k results from the folds then ought to be associated to generate a single prediction.

The intervals for values of the parameters for the SVM models are given in Table 4.1.

Table 4.1 The intervals for values of the parameters for the SVM model

Parameter	**Value Interval**
C	$[2^{-5}, 2^{5}]$
γ	$[2^{-5}, 2^{5}]$
ε	[0, 1]
Degree	[1, 3]

4.2. MLP Model

The MLP architecture depends on the choice of the number of layers, the number of hidden nodes in each of these layers and the objective function. If the number of neurons that is utilized is not sufficient, less information will be acquired. On the contrary, the local minimum might enhance and the network might come close to a local minimum, hence the network's sensitivity will decrease. Nevertheless, there is not a strict regulation for detecting the number of neurons in a hidden layer. Generally, the optimal number is selected with trial and error based on the difficulty of the problem.

In the hidden layer of the MLP models, the tansigmoid function is utilized whereas in the output layer of MLP models, the pure linear function is used. LM algorithm is implemented to train the network. Other important parameters of the MLP based models and their values are given in Table 4.2.

Table 4.2 The intervals for values of the parameters for the MLP models

Parameter	Value Interval
Number of hidden layers	[1, 4]
Number of neurons in hidden layer	[1, 50]
Learning rate	[0, 1]
Momentum	[0, 1]

4.3. RBF Network Model

The performance of the radial basis function (RBF) depends on numerous factor. The choice of basis function and shape parameter have a significant impact on the accuracy of an RBF. The decisions tremendously affect the accuracy and the numerical stability of the method utilized.

Other important parameters of the RBF models are the number of clusters and the clustering seed, and their values are given in Table 4.3.

Table 4.3 The intervals for values of the parameters for the RBF models

Parameter	Value Interval
Number of clusters	[1,4]
Clustering seed	[1,50]

4.4. RF Model

Random forest is an ensemble classifier using many decision tree models in order to improve the classification rate for classification and regression analysis. There are many advantages of random forest such as generating a highly accurate classifier, running efficiently on a large database, giving a prediction about the variable that is important in the classification, having an effective method for estimating missing data, etc. (Ali *et al.,* 2012).

The important parameters that affect the performance of a random forest model are the number of trees, the number of features to be used in random selection and the random number seed. The interval of the parameters used in this study is given in Table 4.4.

Table 4.4 The intervals for values of the parameters for the RF models

Parameter	**Value Interval**
Number of trees in the forest	[1,250]
Number of features	[0,120]
Random number seed	[1,25]

4.5. Holt-Winters Model

The Holt-Winters is a popular statistical forecasting method because it is simple to use, has low data-storage demand, and is easily automated. In this method, the predictive model consists of trended and seasonable patterns that are chosen from noise by averaging historical values. It has some advantages like the easiness of using it, having less computation, and accuracy for seasonal series (Cortez *et al.*, 2012). The seasonal variation can be an additive or multiplicative form. The multiplicative form is used more widely and it has better output results than the additive form. However, there is a limitation that if a data series consist of some values equal to zero, the multiplicative Holt-Winter method cannot be used.

The problem regarding the Holt-Winters method is the selection of smoothing parameters and their initial values, so that prediction better accord with time series data. In this thesis, smoothing factors and initial parameters in Holt-Winters method are estimated by minimizing the prediction result. Hence, the optimal parameters selected with a method in which empirical calculation continued up to lowest prediction error rate met.

The important parameters that affect the performance of a Holt-Winters model are the length of the seasonal cycle, the smoothing factor for the seasonal component, the smoothing factor for the trend and the smoothing factor for the series value. These parameters and their intervals are given in Table 4.5.

Table 4.5 The intervals for values of the parameters for the Holt-Winters model

Parameter	**Value Interval**
Seasonal cycle length	[1,24]
Seasonal smoothing factor	[0,1]
Trend smoothing factor	[0,1]
Value smoothing factor	[0,1]

4.6. Lag Selection

A very important step for time series prediction is the correct selection of the past observations which are named as lags. The lag is an important value for the embedding of the series i.e. for its reconstruction in a state space.

Two methods are usually used to calculate the lags. The first one consists of selecting the first value that corresponds to a zero of the autocorrelation function. The second one selects a value corresponding to a minimum of the mutual information (MI). However, both approaches have the same goal: to select variables that are as much independent (or uncorrelated) as possible in order to reconstruct a trajectory in the state space that approaches at best the true dynamics of the time series. In this study, autocorrelation coefficient calculation is used to select the lags for each time-varying datasets.

4.6.1 Autocorrelation function

The autocorrelation is defined as the correlation of a time series attributes at times *n* among *n-k* where *k=1… K=N-1*. It measures the sign correlated between a time shift and itself where the map of the number of the time shift determined also as a time lag. The autocorrelation function helps to determine the rate of dependence in data. Also, it helps to state stability of time series, utilize likely time series model and sorting sample that occurs again in time series. Further, time series with distinct scale can be compared by using the autocorrelation function.

A time series can be determined by equation

$$X = \{x_{in}: i = 1, \ldots, I; n = 1, \ldots, N\} = \begin{bmatrix} x_{11} & \cdots & x_{i1} & \cdots & x_{I1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{1n} & \cdots & x_{in} & \cdots & x_{In} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{1N} & \cdots & x_{iN} & \cdots & x_{IN} \end{bmatrix}$$

such that $x_n = \{x_{in}: n = 1, \ldots., N\}$ refers the ith time series $(i = 1, \ldots, I), x_{in}$ express the nth investigation $(n = 1, \ldots, N)$ of the ith time series$(i = 1, \ldots, I)$.

For a specific ith time series, $x_n = \{x_{in}: n = 1, \ldots., N\}$ the autocorrelation coefficient at lag $k(k = 1, \ldots, N - 1 = K)$ is shown by

$$\lambda_{ik} = \frac{\sum_{n=k+1}^{N}(x_{in} - \bar{x}_i)\left(x_{i(n-k)} - \bar{x}_i\right)}{\sum_{n-1}^{N}(x_{in} - \bar{x}_i)^2}$$

such that $\bar{x}_i$ is the average of the time i^{th} series. The autocorrelation function of a time series is built by the autocorrelation coefficients at the distinct time lags. The interval of the autocorrelation is [-1,1]. The autocorrelations for each dataset are given in Figure 4.1 through 4.12.

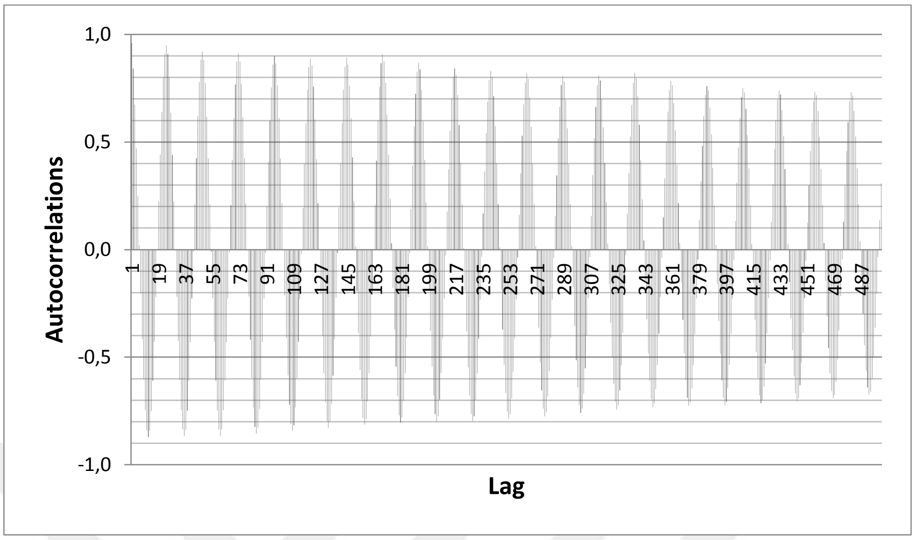

Figure 4.1. Autocorrelations for hourly voice traffic

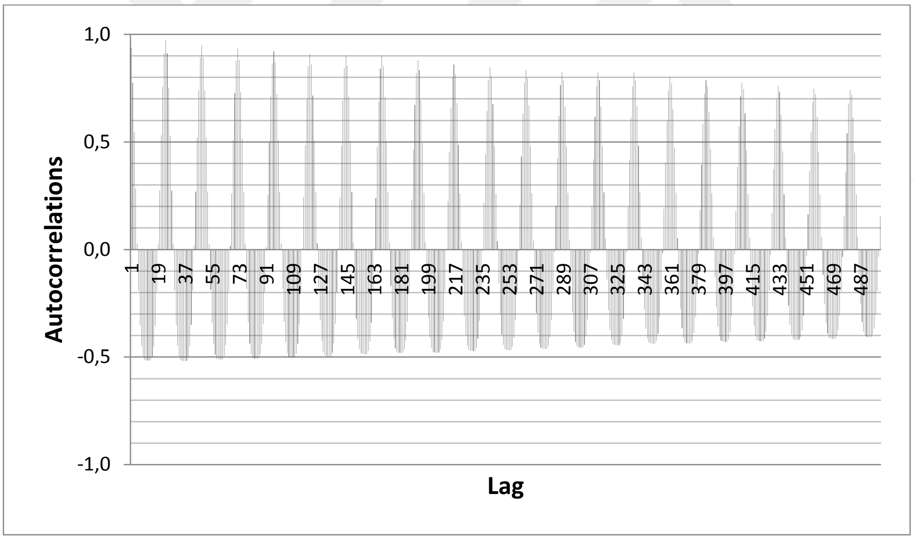

Figure 4.2. Autocorrelations for hourly PS total traffic

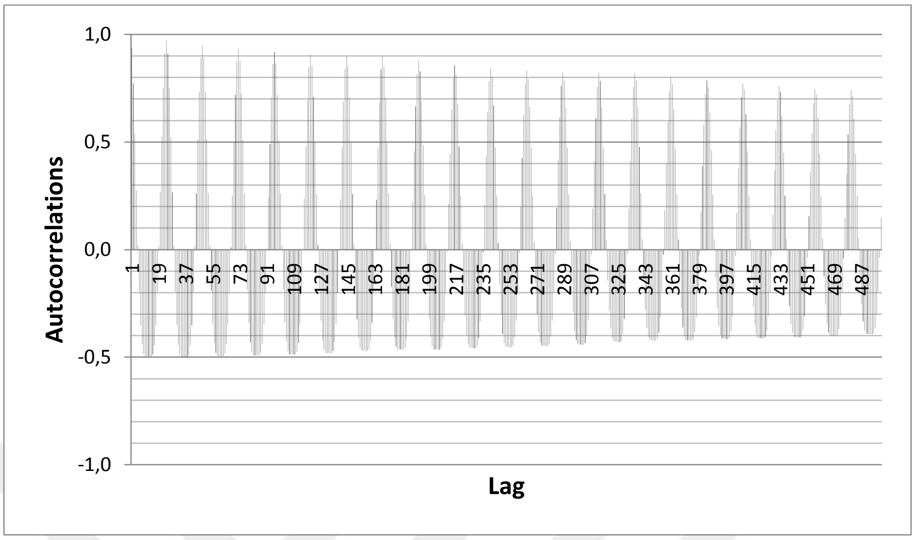

Figure 4.3. Autocorrelations for hourly PS downlink traffic

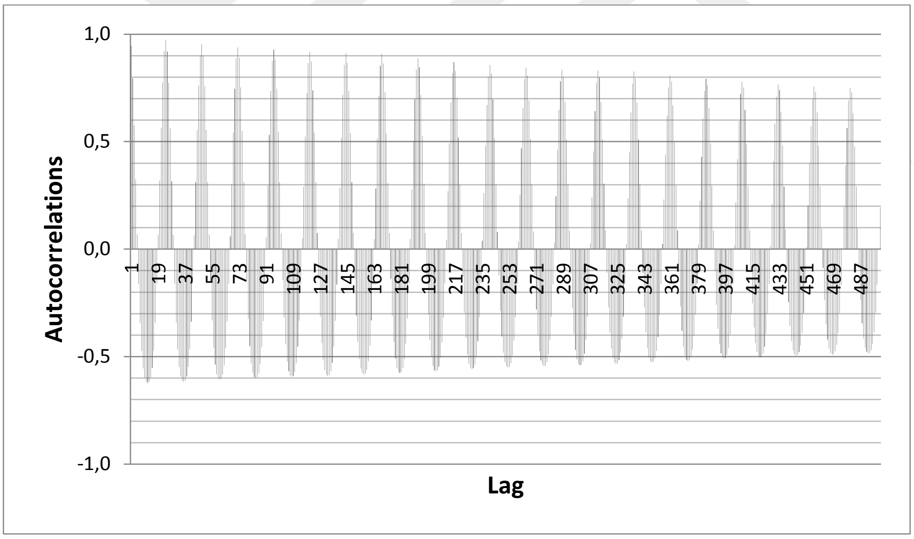

Figure 4.4. Autocorrelations for hourly PS uplink traffic

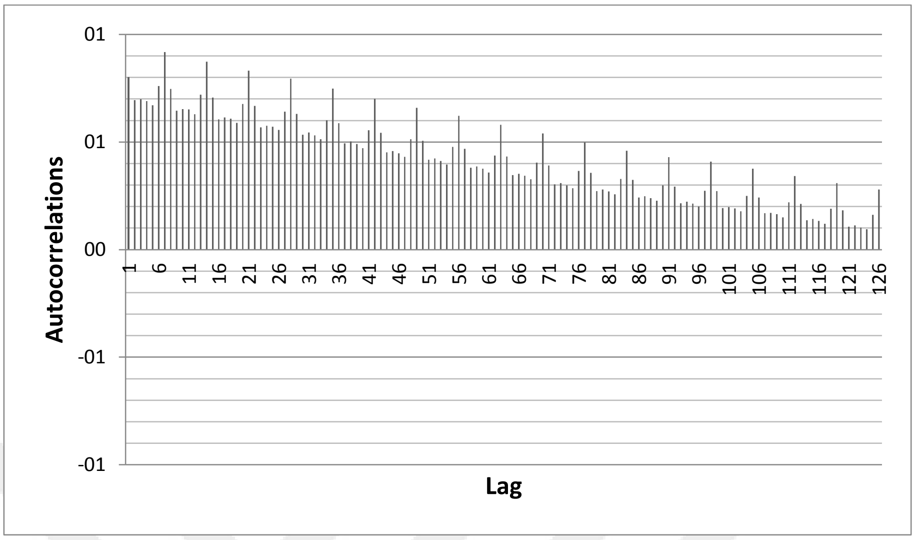

Figure 4.5. Autocorrelations for daily voice traffic

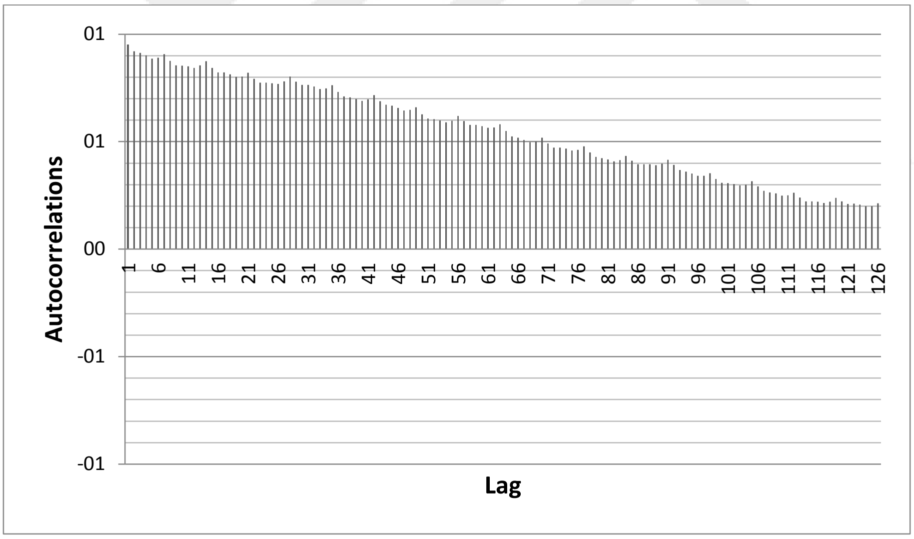

Figure 4.6. Autocorrelations for daily PS total traffic

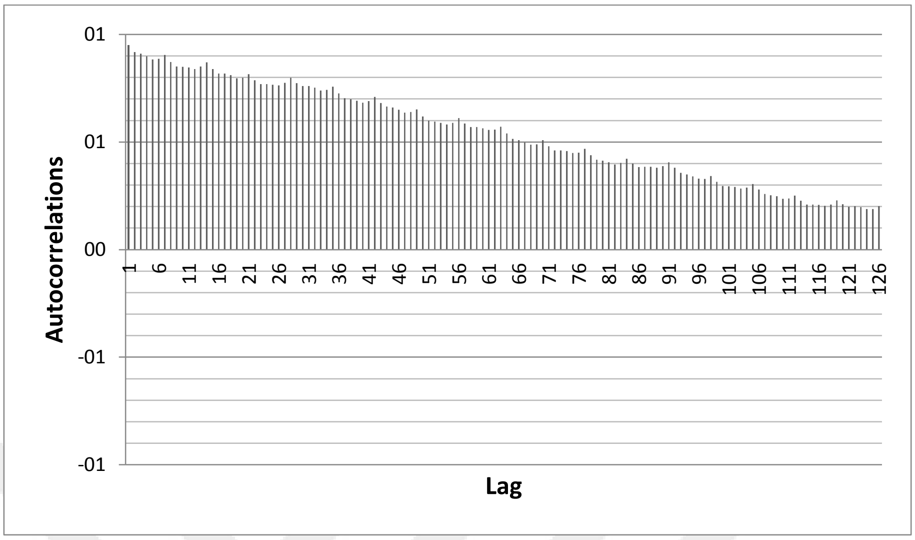

Figure 4.7. Autocorrelations for daily PS downlink traffic

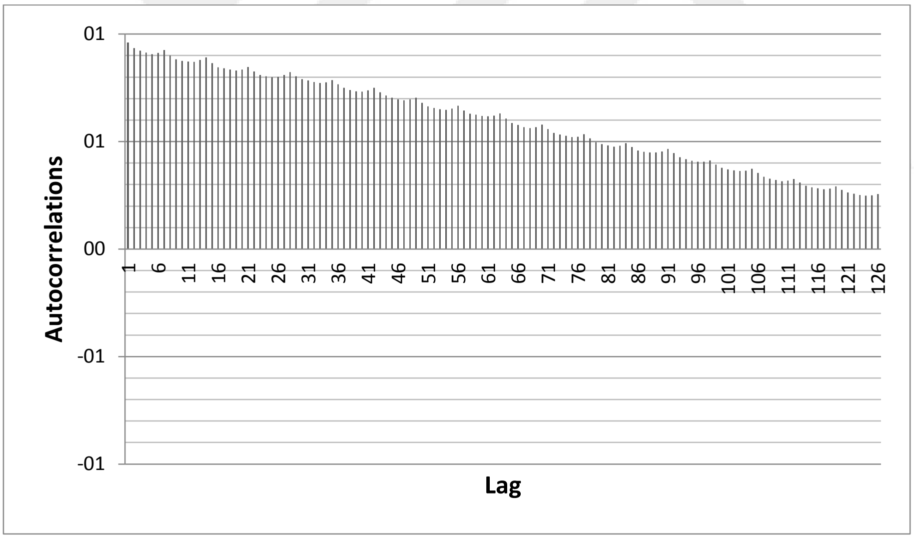

Figure 4.8. Autocorrelations for daily PS uplink traffic

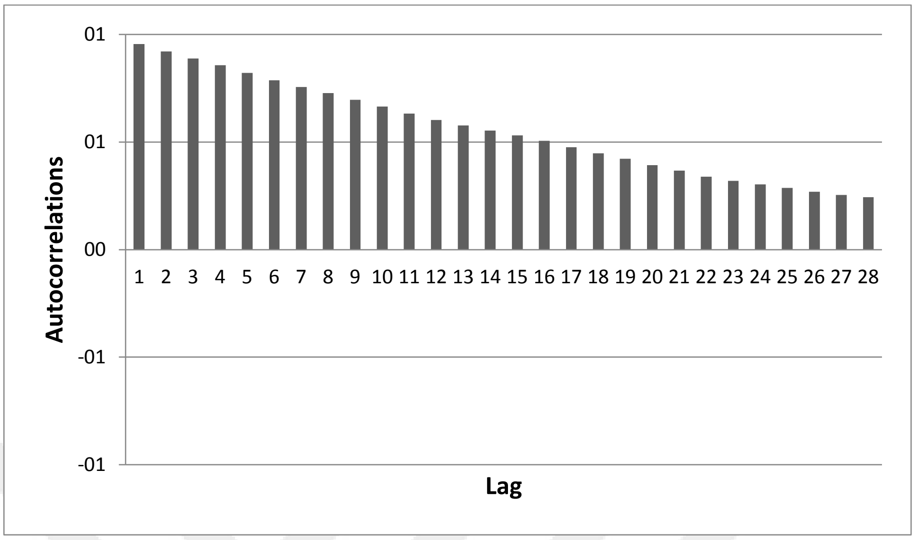

Figure 4.9 Autocorrelations for weekly voice traffic

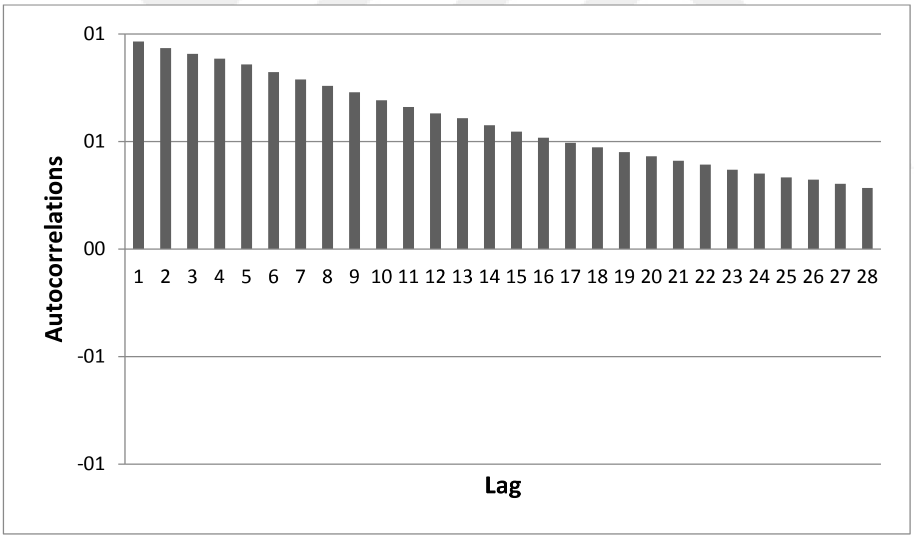

Figure 4.10 Autocorrelations for weekly PS total traffic

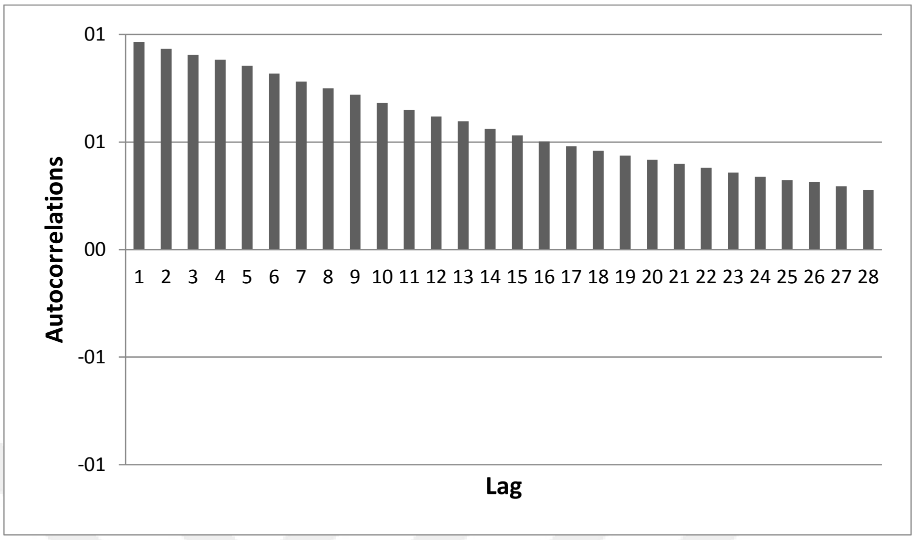

Figure 4.11. Autocorrelations for weekly PS downlink traffic

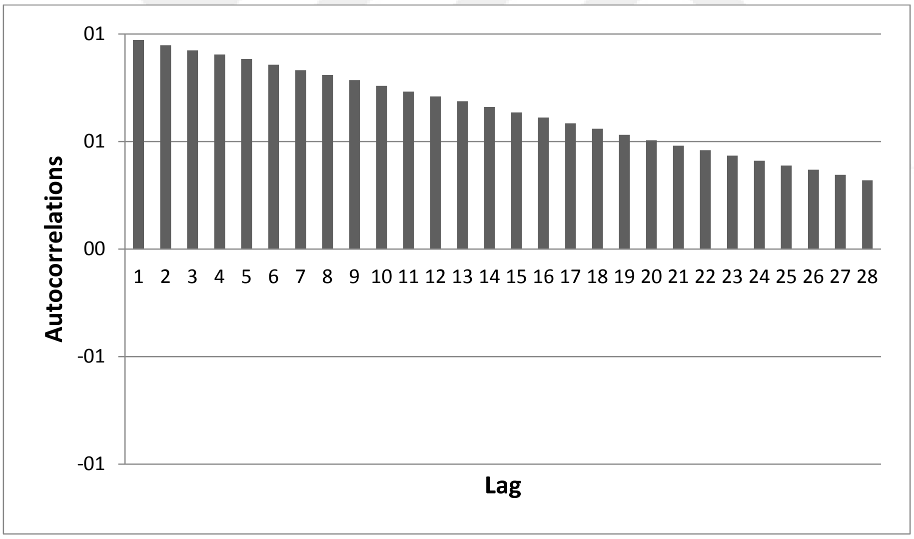

Figure 4.12. Autocorrelations for weekly PS uplink traffic

4.6.2 Time Lag Length Estimation

How many lags should be included in a time series regression is a vital problem to model time series. A small window gives limited information to the network while a large number of time lags can enhance the entropy which has an impact on learning but may increase the forecast error. There are many methods for lag length determination:

a. The F-statistic approach: In order to identify the lag length p, F-statistic test compares the fits of different models.
b. The Bayesian Information Criterion (BIC): To estimate p by minimizing an information criterion

$$BIC = N \ln\left(\frac{SSE}{N}\right) + p \ln(N)$$

such that N defines the number of training, p refers to the number of parameters, SSE means sum squared error. The BIC trades off these two forces so that the number of lags that minimizes the BIC is a consistent estimator of the true lag length.
c. The Akaike information criterion (AIC):

$$AIC = N \ln\left(\frac{SSE}{N}\right) + p \frac{2}{N}$$

In huge number samples, the AIC will overrate p with nonzero probability.

In this thesis, four rules have been utilized to build forecasting models. The rules of the sliding windows are given below;

1. Use all time lags from 1 to a given maximum m: <1, 2… m>
2. Use all lags the autocorrelation values of which are above a given threshold 1.
3. Use all lags the autocorrelation values of which are above a given threshold 2.
4. Use the time lags the autocorrelations values of which are above threshold 3 within the recycling period of the time unit, e.g. recycle period 30 for daily and 169 for the hourly dataset.

By using the heuristic rules for time lag selection, discussed above, four different sliding windows have been generated for each data set.
Table 4.6 through Table 4.17 show the time lags used in each model.

Table 4.6. List of the chosen time lags for hourly voice traffic

Lag Name	**Chosen Lag**
Lag-1	{1,2,….,24}
Lag-2	{1,2,3,4,5}
Lag-3	{1,2,3,4,5,19,20,21,22,23,24,25}
Lag-4	{1,2,3,4,5,19,20,21,22,23,24,25,26,27,28,29,43,44,45,46,47,48,49,50,51,52,53,67,68,69,96,97,98,99,100,101,115,116,117,118,119,146,14,148,149,163,164,165,166,167,168,169}

Table 4.7. List of the chosen time lags for hourly PS total traffic

Lag Name	**Chosen Lag**
Lag-1	{1,2,….,24}
Lag-2	{1,2,3,4}
Lag-3	{1,2,3,4,20,21,22,23,24,25}
Lag-4	{1,2,3,4,20,21,22,23,24,25,26,27,28,44,45,46,47,48,49,50,51,52,68,69,70,71,72,73,74,75,76,92,93,94,95,96,97,98,99,100,116,117,118,119,120,121,122,123,124,140,141,142,143,144,145,146,147,148,164,165,166,167,168,169}

Table 4.8. List of the chosen lags for PS downlink traffic

Lag Name	Chosen Lag
Lag-1	{1,2,…,24}
Lag-2	{1,2,3,4}
Lag-3	{1,2,3,4,20,21,22,23,24,25}
Lag-4	{1,2,3,4,20,21,22,23,24,25,26,27,28,44,45,46,47,48,49,50,51,52,68,69,70,71,72,73,74,75,76,92,93,94,95,96,97,98,99,100,116,117,118,119,120,121,122,123,124,140,141,142,143,144,145,146,147,148,164,165,166,167,168,169}

Table 4.9. List of the chosen lags for hourly PS uplink traffic

Lag Name	Chosen Lag
Lag-1	{1,2…,24}
Lag-2	{1,2,3,4,5}
Lag-3	{1,2,3,4,20,21,22,23,24,25}
Lag-4	{1,2,3,4,5,19,20,21,22,23,24,25,26,27,28,29,43,44,45,46,47,48,49,50,51,52,53,67,68,69,70,71,72,73,74,75,76,77,91,92,93,94,95,96,97,98,99,100,101,115,116,117,118,119,120,121,122,123,124,125,139,140,141,142,143,144,145,146,147,148,149,163,164,165,166,167,168,169}

Table 4.10. List of the chosen time lags for daily voice traffic

Lag Name	Chosen Lag
Lag-1	{1,2,….,30}
Lag-2	{1,6,7,8}
Lag-3	{1,6,7,8,13,14,15,21,28,35,42}
Lag-4	{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30}

Table 4.11. List of the chosen time lags for daily PS total traffic

Lag Name	Chosen Lag
Lag-1	{1,2,….,30}
Lag-2	{1,6,7,8}
Lag-3	{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,28}
Lag-4	{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25, 26,27,28,29,30}

Table 4.12. List of the chosen time lags for daily PS downlink traffic

Lag Name	Chosen Lag
Lag-1	{1,2,….,30}
Lag-2	{1,2,3,4,5,6,7}
Lag-3	{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18}
Lag-4	{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25, 26,27,28,29,30}

Table 4.13. List of the chosen time lags for daily PS uplink traffic

Lag Name	Chosen Lag
Lag-1	{1,2,….,30}
Lag-2	{1,2,3,4,5,6,7}
Lag-3	{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18}
Lag-4	{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25, 26,27,28,29,30}

Table 4.14. List of chosen time lags for weekly voice traffic

Lag Name	Chosen Lag
Lag-1	{1,2,….,52}
Lag-2	{1,2}
Lag-3	{1,2,3,4}
Lag-4	{1,2,3,4,5,6,7,8,9,10,11,12}

Table 4.15. List of chosen time lags for weekly PS total traffic

Lag Name	Chosen Lag
Lag-1	{1,2,….,52}
Lag-2	{1,2,3}
Lag-3	{1,2,3,4,5,6}
Lag-4	{1,2,3,4,5,6,7,8,9,10,11,12}

Table 4.16. List of the chosen time lags for weekly PS downlink traffic

Lag Name	Chosen Lag
Lag-1	{1,2,….,52}
Lag-2	{1,2,3}
Lag-3	{1,2,3,4,5,6}
Lag-4	{1,2,3,4,5,6,7,8,9,10,11,12}

Table 4.17. List of the chosen time lags for weekly PS uplink traffic

Lag Name	Chosen Lag
Lag-1	{1,2,….,52}
Lag-2	{1,2,3,4}
Lag-3	{1,2,3,4,5,6,7,8}
Lag-4	{1,2,3,4,5,6,7,8,9,10,11,12}

4.7. Performance Metric

The performance of the forecasting models have been evaluated by computing Mean Absolute Percentage Error (MAPE) value which is a metric used commonly in forecasting applications. It has an advantage of being scale independent, so it is frequently usable for the comparison of forecast performance between different series.

The formula of MAPE is

$$MAPE = \frac{100}{n}\sum_{t=1}^{n}\frac{|A_t - F_t|}{A_t}$$

where n is the number of the forecast, A_t is the actual value, and F_t is the forecast value (Benzer *et al.*, 2015).

5. RESULTS AND DISCUSSION

In this thesis, five different methods including SVM, MLP, RBF, RF, and Holt-Winters have been employed to forecast the circuit switched voice and packet switched data traffic for live UMTS network. In order to build models, three different train/test split combination have been utilized. The split ranges are 70% for training and 30% for testing, 80% for training 20% for testing, and 90% for training and 10% for testing, respectively.

Table 5.1 through Table 5.12 show the computed *MAPE*'s for hourly traffic dataset.

Table 5.1. *MAPE* values for hourly CS voice traffic prediction with 70%-30% split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Hourly CS traffic lag-1	8.61	9.83	11.96	80.33	22.18
Hourly CS traffic lag-2	14.52	19.63	11.47	187.41	22.18
Hourly CS traffic lag-3	7.84	11.11	11.97	132.33	22.18
Hourly CS traffic lag-4	7.82	10.60	11.77	36.71	9.29

Table 5.2. *MAPE* values for hourly PS total traffic prediction with *70%-30%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Hourly PS total traffic lag-1	3.51	4.76	4.03	18.06	8.78
Hourly PS total traffic lag-2	5.76	7.08	3.97	30.56	8.78
Hourly PS total traffic lag-3	3.41	5.30	4.39	22.45	8.78
Hourly PS total traffic lag-4	3.32	5.16	4.38	15.68	6.86

Table 5.3. *MAPE* values for hourly PS downlink traffic prediction with *70%-30%*split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Hourly PS downlink traffic lag-1	3.49	3.68	4.00	57.59	9.01
Hourly PS downlink traffic lag-2	5.76	7.33	3.83	68.04	9.01
Hourly PS downlink traffic lag-3	3.48	3.50	4.48	41.25	9.01
Hourly PS downlink traffic lag-4	2.97	3.31	4.41	14.27	6.92

Table 5.4. *MAPE* values for hourly PS uplink traffic prediction with *70%-30%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Hourly PS uplink traffic lag-1	3.63	4.23	4.23	38.92	9.87
Hourly PS uplink traffic lag-2	6.47	4.78	4.27	40.31	9.87
Hourly PS uplink traffic lag-3	3.54	4.03	4.32	18.67	9.87
Hourly PS uplink traffic lag-4	3.25	4.75	3.95	15.35	8.86

Table 5.5. *MAPE* values for hourly CS voice traffic prediction with *80%-20%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Hourly CS voice traffic lag-1	7.13	7.59	7.72	25.17	22.03
Hourly CS voice traffic lag-2	13.76	9.74	8.22	41.15	20.15
Hourly CS voice traffic lag-3	6.80	9.55	7.88	32.65	20.15
Hourly CS voice traffic lag-4	5.32	8.93	7.74	24.36	6.88

Table 5.6. *MAPE* values for hourly PS total traffic prediction with *80%-20%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Hourly PS total traffic lag-1	3.13	3.67	3.27	12.84	10.73
Hourly PS total traffic lag-2	5.80	4.49	2.94	17.69	7.87
Hourly PS total traffic lag-3	3.14	3.52	3.45	12.10	7.87
Hourly PS total traffic lag-4	2.51	3.34	3.56	12.05	4.77

Table 5.7. *MAPE* values for hourly PS downlink traffic prediction with *80%-20%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Hourly PS downlink traffic lag-1	3.15	3.37	3.33	12.69	8.07
Hourly PS downlink traffic lag-2	5.63	4.60	2.93	15.89	8.49
Hourly PS downlink traffic lag-3	3.18	3.27	3.36	11.95	8.07
Hourly PS downlink traffic lag-4	2.60	3.47	3.63	11.66	6.20

Table 5.8. *MAPE* values for hourly PS uplink traffic prediction with *80%-20%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Hourly PS uplink traffic lag-1	3.44	3.80	4.46	11.26	8.75
Hourly PS uplink traffic lag-2	7.14	4.67	3.67	18.53	6.40
Hourly PS uplink traffic lag-3	3.13	3.50	4.75	13.33	6.60
Hourly PS uplink traffic lag-4	2.62	4.24	4.99	10.84	6.38

Table 5.9. *MAPE* values for hourly CS voice traffic prediction with *90%-10%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Hourly CS voice traffic lag-1	6.59	6.89	7.51	21.85	17.58
Hourly CS voice traffic lag-2	11.35	8.72	7.55	49.19	4.48
Hourly CS voice traffic lag-3	6.50	9.36	7.42	23.34	4.52
Hourly CS voice traffic lag-4	3.31	6.75	7.39	16.06	4.48

Table 5.10. *MAPE* values for hourly PS total traffic prediction with *90%-10%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Hourly PS total traffic lag-1	2.89	2.89	3.34	9.50	8.18
Hourly PS total traffic lag-2	3.19	3.19	2.93	10.38	3.96
Hourly PS total traffic lag-3	3.24	2.77	3.39	9.77	3.96
Hourly PS total traffic lag-4	2.10	2.95	3.43	9.20	3.67

Table 5.11. *MAPE* values for hourly PS downlink traffic prediction with *90%-10%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Hourly PS downlink traffic lag-1	2.94	2.92	3.29	10.04	7.23
Hourly PS downlink traffic lag-2	3.58	3.20	2.96	12.45	7.30
Hourly PS downlink traffic lag-3	2.90	2.77	3.50	10.31	7.29
Hourly PS downlink traffic lag-4	2.11	3.31	2.65	8.49	2.06

Table 5.12. *MAPE* values for hourly PS uplink traffic prediction with *90%-10%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Hourly PS uplink traffic lag-1	2.95	2.84	2.82	8.09	7.81
Hourly PS uplink traffic lag-2	4.14	2.87	2.29	10.72	6.42
Hourly PS uplink traffic lag-3	3.26	2.75	2.74	10.34	7.81
Hourly PS uplink traffic lag-4	1.95	2.85	2.68	7.85	2.03

Table 5.13 through Table 5.24 show the *MAPE*'s of the forecasting models in each category for daily data set with a pre-defined split rate, separately.

Table 5.13. *MAPE* values for daily CS voice traffic prediction with *70%-30%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Daily CS voice traffic lag-1	7.63	5.90	15.86	11.24	3.38
Daily CS voice traffic lag-2	4.00	5.93	16.71	18.93	3.38
Daily CS voice traffic lag-3	3.52	6.31	14.46	16.97	3.39
Daily CS voice traffic lag-4	3.64	4.87	16.95	17.04	3.36

Table 5.14. *MAPE* values for daily PS total traffic prediction with *70%-30%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Daily PS total traffic lag-1	3.75	3.49	12.91	17.51	5.09
Daily PS total traffic lag-2	4.48	3.76	12.89	12.92	5.09
Daily PS total traffic lag-3	3.74	5.25	12.25	18.49	5.09
Daily PS total traffic lag-4	3.43	6.09	11.96	17.96	3.14

Table 5.15. *MAPE* values for daily PS downlink traffic prediction with *70%-30%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Daily PS downlink traffic lag-1	2.60	3.89	13.80	10.50	5.32
Daily PS downlink traffic lag-2	2.60	3.89	13.80	10.50	5.32
Daily PS downlink traffic lag-3	2.18	4.62	14.84	13.59	5.32
Daily PS downlink traffic lag-4	2.19	5.05	13.96	14.56	3.31

Table 5.16. *MAPE* values for daily PS uplink traffic prediction with *70%-30%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Daily PS uplink traffic lag-1	2.60	3.89	5.00	6.84	4.25
Daily PS uplink traffic lag-2	2.60	3.89	5.00	6.84	4.25
Daily PS uplink traffic lag-3	2.18	5.60	5.34	7.7	4.13
Daily PS uplink traffic lag-4	2.20	5.06	4.08	8.73	2.82

Table 5.17. *MAPE* values for daily CS voice traffic prediction with *80%-20%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Daily CS voice traffic lag-1	2.67	3.58	11.99	10.21	2.90
Daily CS voice traffic lag-2	2.67	3.58	12.25	11.97	2.42
Daily CS voice traffic lag-3	2.34	4.76	11.72	10.58	2.42
Daily CS voice traffic lag-4	2.25	4.65	11.24	10.55	2.42

Table 5.18. *MAPE* values for daily PS total traffic prediction with *80%-20%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Daily PS total traffic lag-1	2.71	5.43	5.00	10.96	5.97
Daily PS total traffic lag-2	2.71	5.13	4.82	10.10	3.12
Daily PS total traffic lag-3	2.59	4.41	4.35	9.72	3.12
Daily PS total traffic lag-4	2.89	3.99	4.55	7.26	3.12

Table 5.19. *MAPE* values for daily PS downlink traffic prediction with *80%-20%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Daily PS downlink traffic lag-1	2.85	4.84	5.18	12.06	4.96
Daily PS downlink traffic lag-2	2.85	4.84	5.18	12.07	4.48
Daily PS downlink traffic lag-3	2.73	4.60	5.21	11.18	3.30
Daily PS downlink traffic lag-4	2.97	4.24	4.99	7.89	3.30

Table 5.20. *MAPE* values for daily PS uplink traffic prediction with *80%-20%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Daily PS uplink traffic lag-1	2.02	2.77	2.81	4.08	4.16
Daily PS uplink traffic lag-2	2.02	2.77	2.81	4.08	2.34
Daily PS uplink traffic lag-3	2.03	3.59	2.87	3.94	2.34
Daily PS uplink traffic lag-4	2.15	2.87	3.15	3.75	2.34

Table 5.21. *MAPE* values for daily CS voice traffic prediction with *90%-10%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Daily CS voice traffic lag-1	3.11	4.24	5.28	9.40	3.23
Daily CS voice traffic lag-2	3.15	2.78	5.32	9.52	2.73
Daily CS voice traffic lag-3	3.11	3.05	5.63	8.98	2.59
Daily CS voice traffic lag-4	2.93	4.75	5.47	8.93	2.59

Table 5.22. *MAPE* values for daily PS total traffic prediction with *90%-10%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Daily PS total traffic lag-1	3.02	3.66	4.82	4.73	7.06
Daily PS total traffic lag-2	2.94	3.50	4.91	4.78	3.30
Daily PS total traffic lag-3	3.30	3.89	4.55	4.80	3.30
Daily PS total traffic lag-4	3.59	4.01	4.45	4.82	3.22

Table 5.23. *MAPE* values for daily PS downlink traffic prediction with *90%-10%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Daily PS downlink traffic lag-1	3.07	3.87	5.37	5.31	6.89
Daily PS downlink traffic lag-2	3.09	3.87	5.37	5.76	3.44
Daily PS downlink traffic lag-3	3.38	3.88	5.20	5.18	3.44
Daily PS downlink traffic lag-4	3.73	4.07	4.61	4.93	3.44

Table 5.24. *MAPE* values for daily PS uplink traffic prediction with *90%-10%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Daily PS uplink traffic lag-1	2.40	2.21	2.98	3.87	3.67
Daily PS uplink traffic lag-2	2.40	2.21	2.98	3.87	2.57
Daily PS uplink traffic lag-3	2.56	2.43	3.09	3.67	2.57
Daily PS uplink traffic lag-4	2.82	3.14	3.49	3.98	2.57

Table 5.25 through Table 5.36 show the *MAPE*'s of the forecasting models in each category for weekly data set with a pre-defined split rate, separately.

Table 5.25. *MAPE* values for weekly CS voice traffic prediction with *70%-30%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Weekly CS voice traffic lag-1	7.63	5.90	15.86	11.24	3.38
Weekly CS voice traffic lag-2	4.00	5.93	16.71	18.93	3.38
Weekly CS voice traffic lag-3	3.52	6.31	14.46	16.97	3.39
Weekly CS voice traffic lag-4	3.64	4.87	16.95	17.04	3.36

Table 5.26 *MAPE* values for weekly PS total traffic prediction with *70%-30%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Weekly PS total traffic lag-1	3.54	9.70	16.27	13.66	3.18
Weekly PS total traffic lag-2	3.45	7.90	27.46	19.70	3.17
Weekly PS total traffic lag-3	3.43	8.41	21.27	13.67	3.17
Weekly PS total traffic lag-4	3.32	7.88	19.55	15.38	3.10

Table 5.27. *MAPE* values for weekly PS downlink traffic prediction with *70%-30%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Weekly PS downlink traffic lag-1	2.60	3.89	13.80	10.50	5.32
Weekly PS downlink traffic lag-2	2.60	3.89	13.80	10.50	5.32
Weekly PS downlink traffic lag-3	2.18	4.62	14.84	13.59	5.32
Weekly PS downlink traffic lag-4	2.19	5.05	13.96	14.56	3.31

Table 5.28. *MAPE* values for weekly PS uplink traffic prediction with *70%-30%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Weekly PS uplink traffic lag-1	2.60	3.89	5.00	6.84	4.25
Weekly PS uplink traffic lag-2	2.60	3.89	5.00	6.84	4.25
Weekly PS uplink traffic lag-3	2.18	5.60	5.34	7.7	4.13
Weekly PS uplink traffic lag-4	2.20	5.06	4.08	8.73	2.82

Table 5.29. *MAPE* values for weekly CS voice traffic prediction with *80%-20%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Weekly CS voice traffic lag-1	2.67	3.58	11.99	10.21	2.90
Weekly CS voice traffic lag-2	2.67	3.58	12.25	11.97	2.42
Weekly CS voice traffic lag-3	2.34	4.76	11.72	10.58	2.42
Weekly CS voice traffic lag-4	2.25	4.65	11.24	10.55	2.42

Table 5.30. *MAPE* values for weekly PS total traffic prediction with *80%-20%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Weekly PS total traffic lag-1	2.71	5.43	5.00	10.96	5.97
Weekly PS total traffic lag-2	2.71	5.13	4.82	10.10	3.12
Weekly PS total traffic lag-3	2.59	4.41	4.35	9.72	3.12
Weekly PS total traffic lag-4	2.89	3.99	4.55	7.26	3.12

Table 5.31. *MAPE* values for weekly PS downlink traffic prediction with *80%-20%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Weekly PS downlink traffic lag-1	2.85	4.84	5.18	12.06	4.96
Weekly PS downlink traffic lag-2	2.85	4.84	5.18	12.07	4.48
Weekly PS downlink traffic lag-3	2.73	4.60	5.21	11.18	3.30
Weekly PS downlink traffic lag-4	2.97	4.24	4.99	7.89	3.30

Table 5.32. *MAPE* values for weekly PS uplink traffic prediction with *80%-20%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Weekly PS uplink traffic lag-1	2.02	2.77	2.81	4.08	4.16
Weekly PS uplink traffic lag-2	2.02	2.77	2.81	4.08	2.34
Weekly PS uplink traffic lag-3	2.03	3.59	2.87	3.94	2.34
Weekly PS uplink traffic lag-4	2.15	2.87	3.15	3.75	2.34

Table 5.33. *MAPE* values for weekly CS voice traffic prediction with *90%-10%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Weekly CS voice traffic lag-1	3.11	4.24	5.28	9.40	3.23
Weekly CS voice traffic lag-2	3.15	2.78	5.32	9.52	2.73
Weekly CS voice traffic lag-3	3.11	3.05	5.63	8.98	2.59
Weekly CS voice traffic lag-4	2.93	4.75	5.47	8.93	2.59

Table 5.34. *MAPE* values for weekly PS total traffic prediction with *90%-10%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Weekly PS total traffic lag-1	3.02	3.66	4.82	4.73	7.06
Weekly PS total traffic lag-2	2.94	3.50	4.91	4.78	3.30
Weekly PS total traffic lag-3	3.30	3.89	4.55	4.80	3.30
Weekly PS total traffic lag-4	3.59	4.01	4.45	4.82	3.22

Table 5.35. *MAPE* values for weekly PS downlink traffic prediction with *90%-10%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Weekly PS downlink traffic lag-1	3.07	3.87	5.37	5.31	6.89
Weekly PS downlink traffic lag-2	3.09	3.87	5.37	5.76	3.44
Weekly PS downlink traffic lag-3	3.38	3.88	5.20	5.18	3.44
Weekly PS downlink traffic lag-4	3.73	4.07	4.61	4.93	3.44

Table 5.36. *MAPE* values for weekly PS uplink traffic prediction with *90%-10%* split rate

Models	SVM	MLP	RF	RBF	Holt-Winters
Weekly PS uplink traffic lag-1	2.40	2.21	2.98	3.87	3.67
Weekly PS uplink traffic lag-2	2.40	2.21	2.98	3.87	2.57
Weekly PS uplink traffic lag-3	2.56	2.43	3.09	3.67	2.57
Weekly PS uplink traffic lag-4	2.82	3.14	3.49	3.98	2.57

Figure 5.1 through Figure 5.12 show the average of the *MAPE*'s of all methods separately and the percentage decrement rates in *MAPE*'s between the models having the lowest *MAPE*'s on the average and the other regression models for each data set.

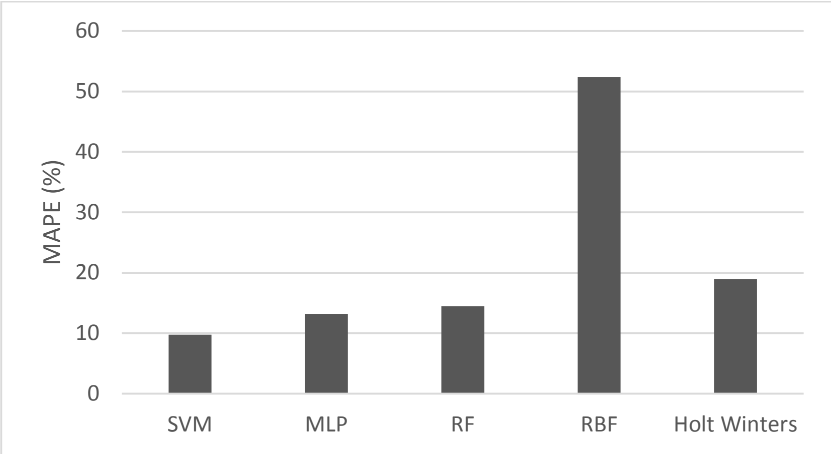

Figure 5.1. Average *MAPE* of the forecasting models for hourly CS voice traffic with *70%-30%* split rate

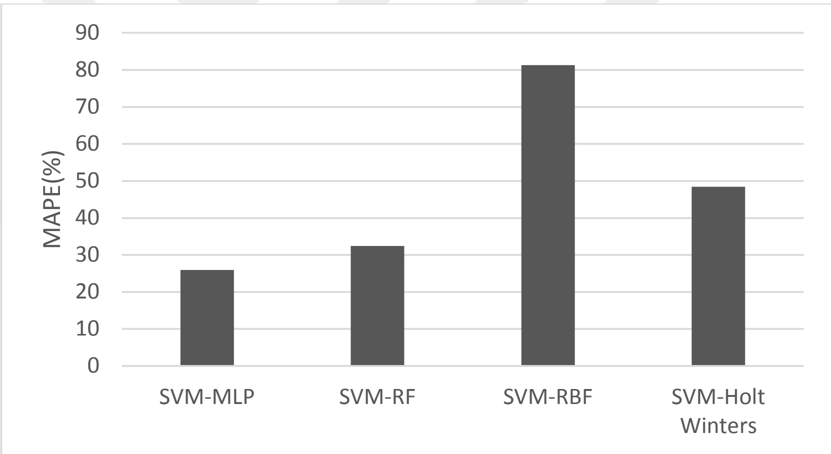

Figure 5.2. Percentage decrease rates in average *MAPE* of the forecasting models for hourly CS voice traffic with *70%-30%* split rate

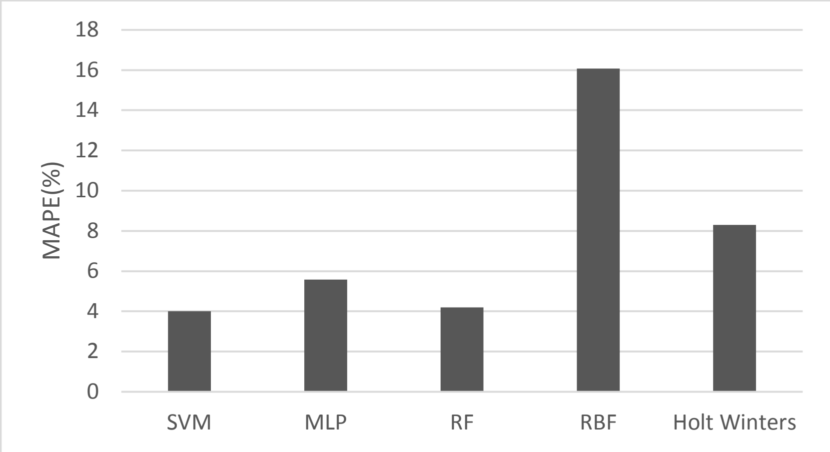

Figure 5.3. Average *MAPE* of the forecasting models for hourly PS total traffic with *70%-30%* split rate

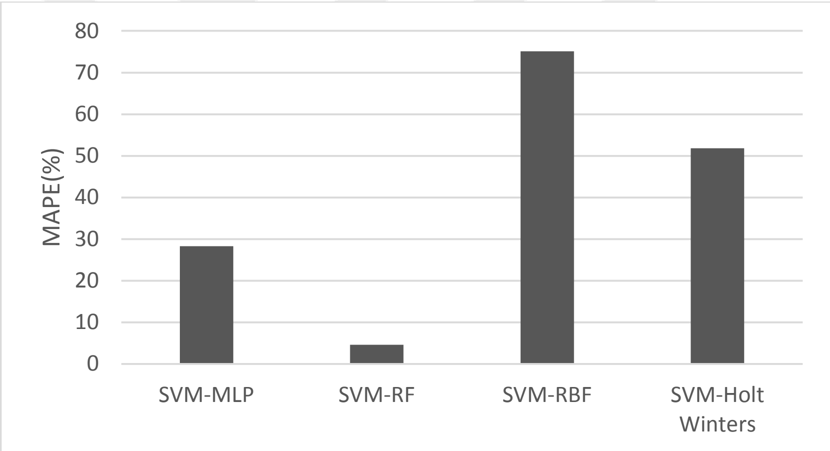

Figure 5.4. Percentage decrease rates in average *MAPE* of the forecasting models for hourly PS total traffic with *70%-30%* split rate

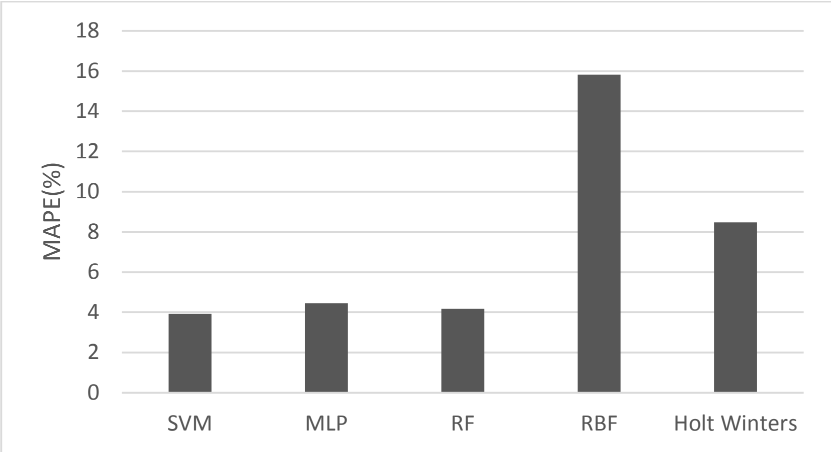

Figure 5.5. Average *MAPE* of the forecasting models for hourly PS downlink traffic with *70%-30%* split rate

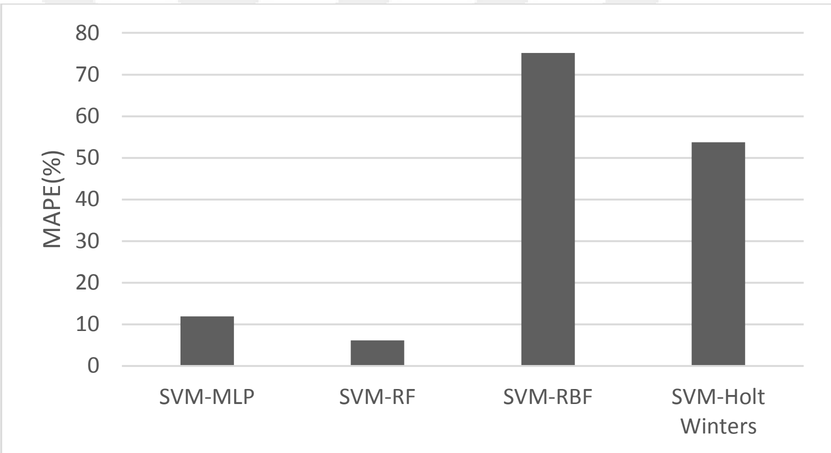

Figure 5.6. Percentage decrease rates in average *MAPE* of the forecasting models for hourly PS downlink traffic with *70%-30%* split rate

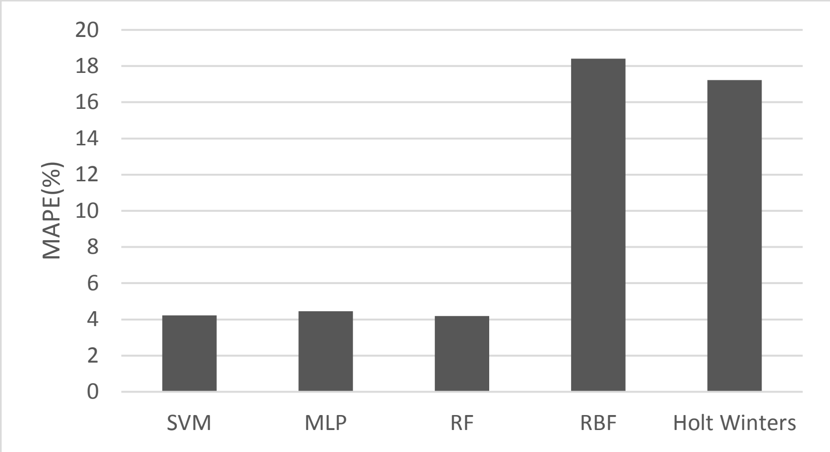

Figure 5.7. Average *MAPE* of the forecasting models for hourly PS uplink traffic with *70%-30%* split rate

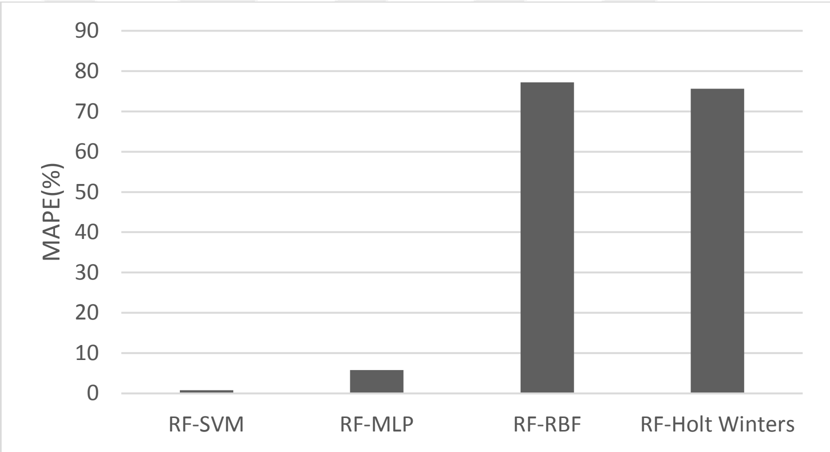

Figure 5.8. Percentage decrease rates in average *MAPE* of the forecasting models for hourly PS uplink traffic with *70%-30%* split rate

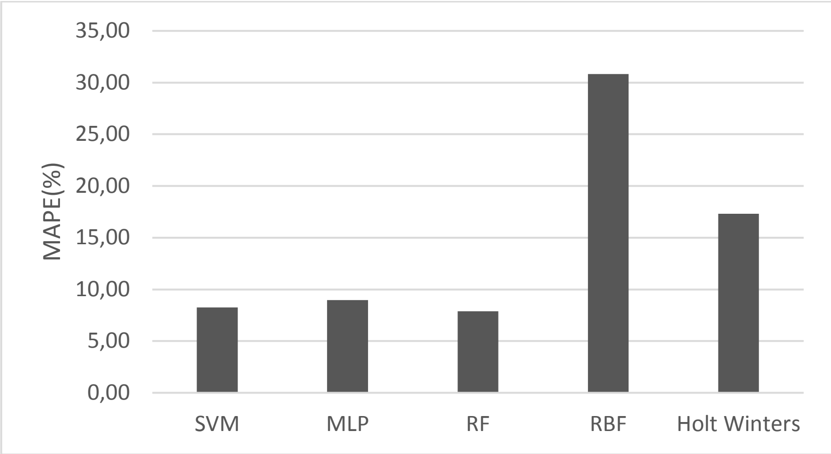

Figure 5.9. Average *MAPE* of the forecasting models for hourly CS voice traffic with *80%-20%* split rate

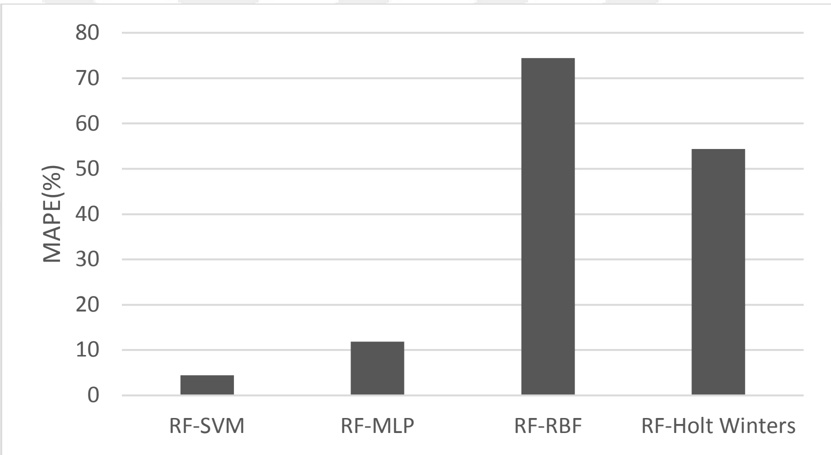

Figure 5.10. Percentage decrease rates in average *MAPE* of the forecasting models for hourly CS voice traffic with *80%-20%* split rate

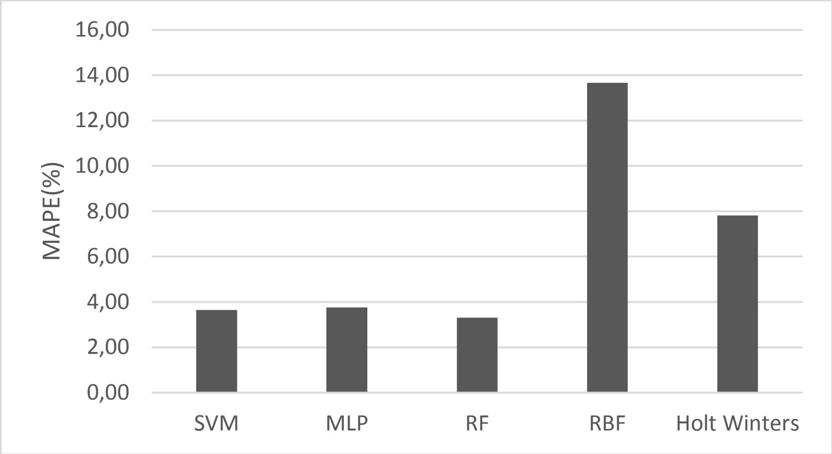

Figure 5.11. Average *MAPE* of the forecasting models for hourly PS total traffic with *80%-20%* split rate

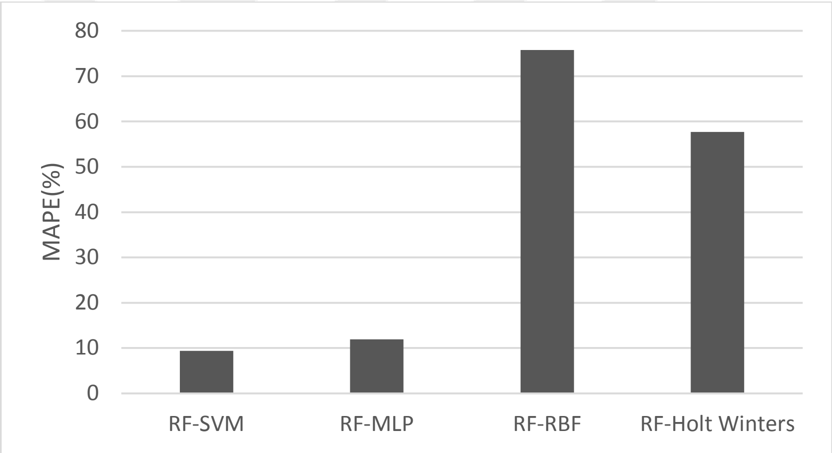

Figure 5.12. Percentage decrease rates in average *MAPE* of the forecasting models for hourly PS total traffic with *80%-20%* split rate

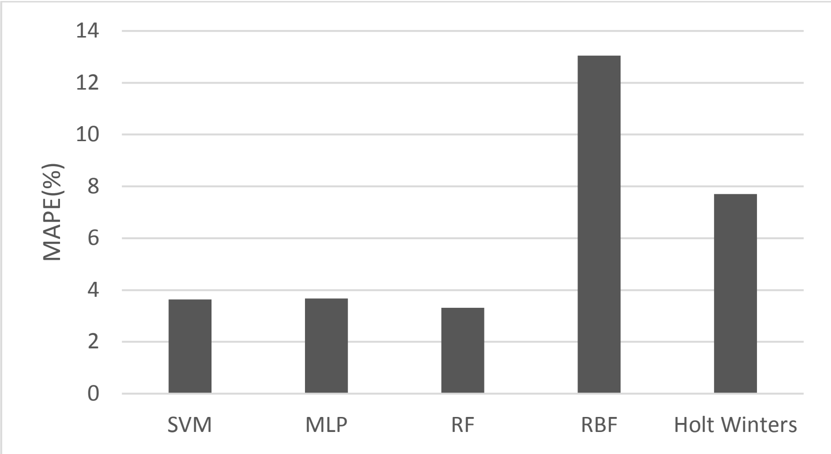

Figure 5.13. Average *MAPE* of the forecasting models for hourly PS downlink traffic with *80%-20%* split rate

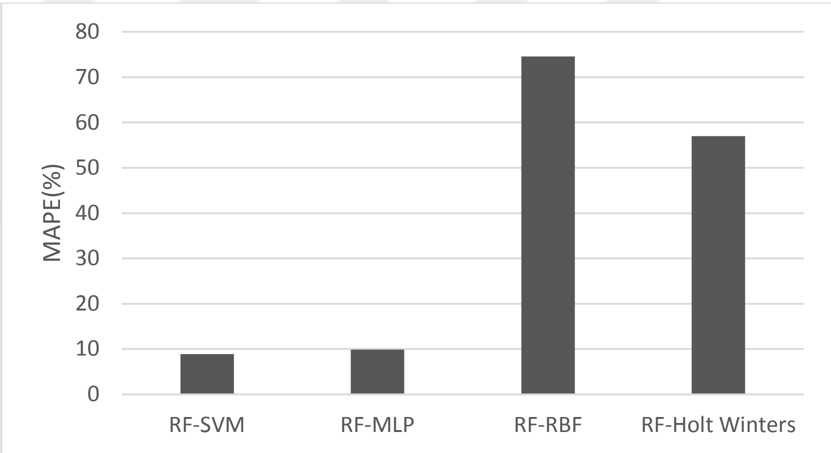

Figure 5.14. Percentage decrease rates in average *MAPE* of the forecasting models for hourly PS downlink traffic with *80%-20%* split rate

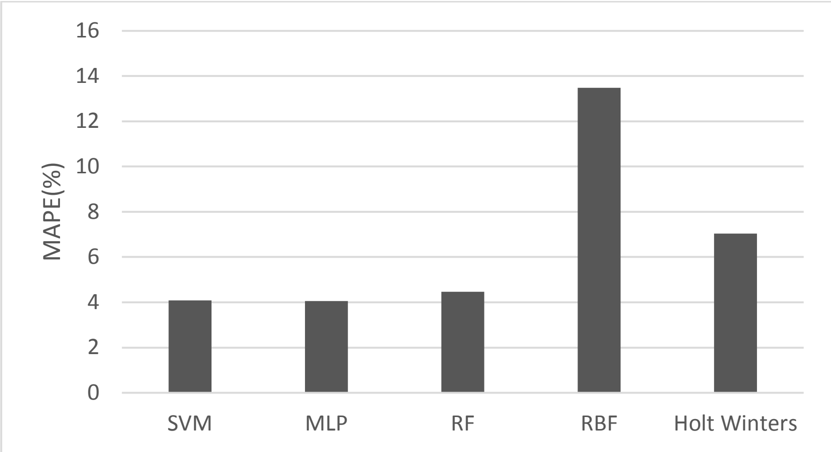

Figure 5.15. Average *MAPE* of the forecasting models for hourly PS uplink traffic with *80%-20%* split rate

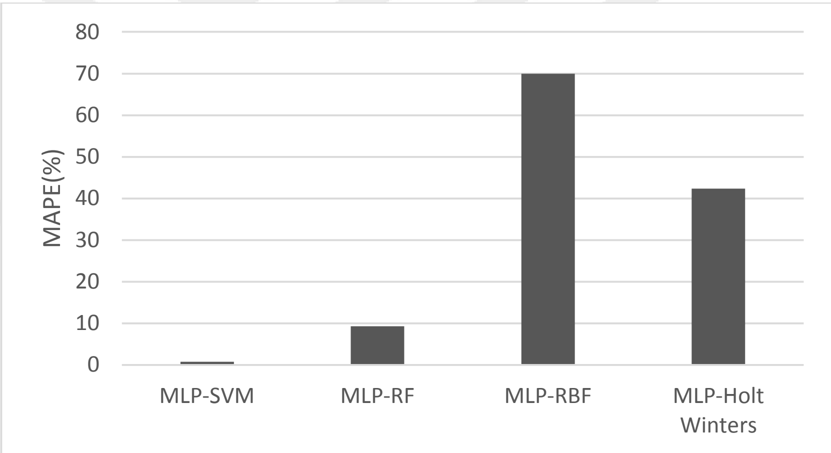

Figure 5.16. Percentage decrease rates in average *MAPE* of the forecasting models for hourly PS uplink traffic with *80%-20%* split rate

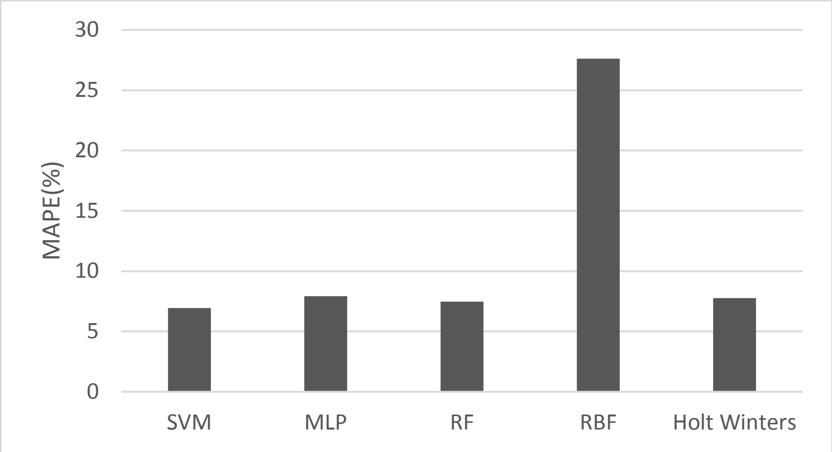

Figure 5.17. Average *MAPE* of the forecasting models for hourly CS voice traffic with *90%-10%* split rate

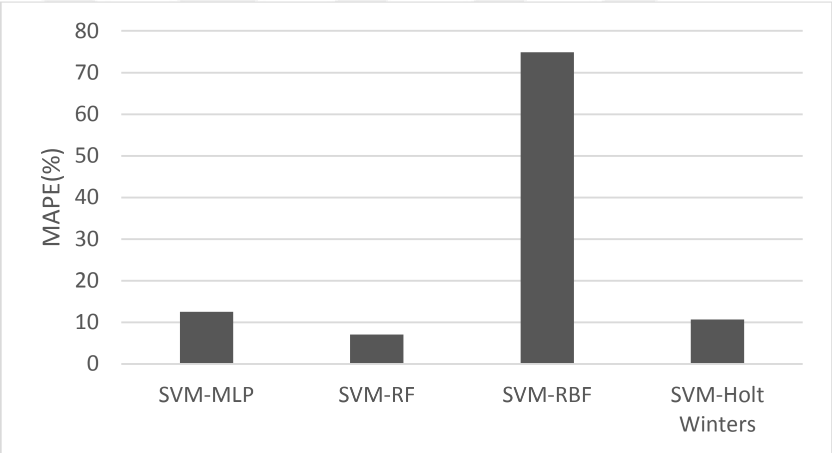

Figure 5.18. Percentage decrease rates in average *MAPE* of the forecasting models for hourly CS voice traffic with *90%-10%* split rate

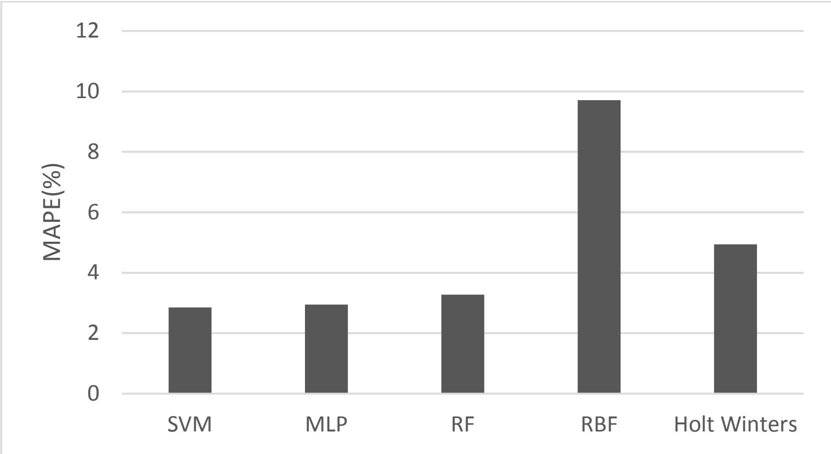

Figure 5.19. Average *MAPE* of the forecasting models for hourly PS total traffic with *90%-10%* split rate

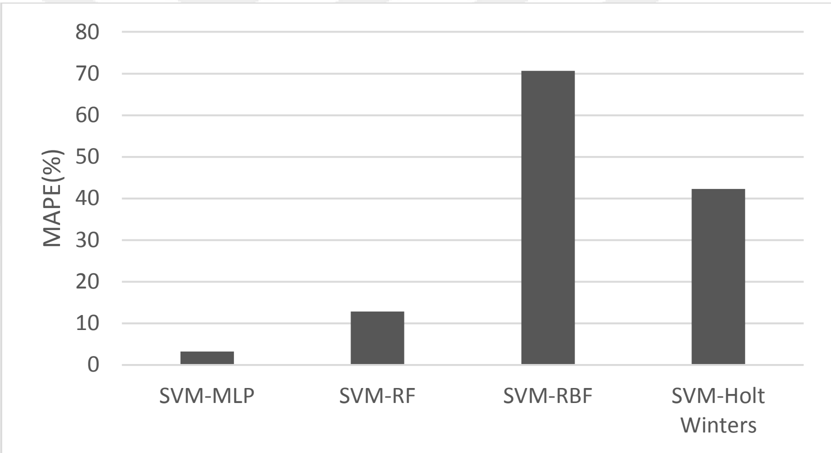

Figure 5.20. Percentage decrease rates in average *MAPE* of the forecasting models for hourly PS total traffic with *90%-10%* split rate

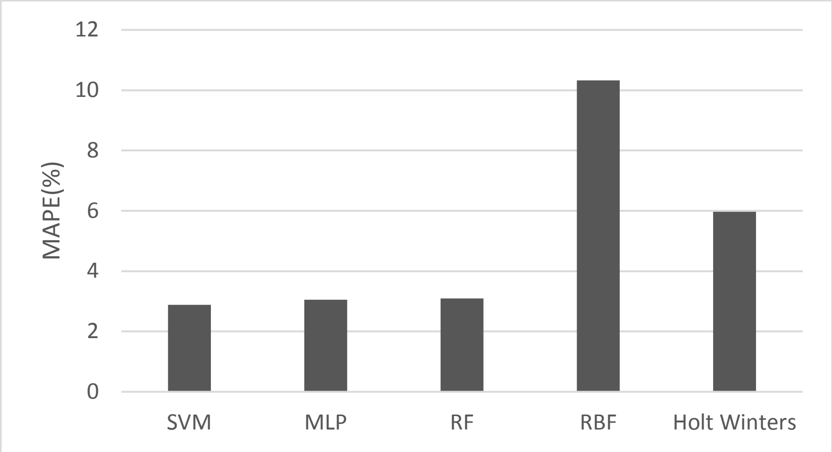

Figure 5.21. Average *MAPE* of the forecasting models for hourly PS downlink traffic with *90%-10%* split rate

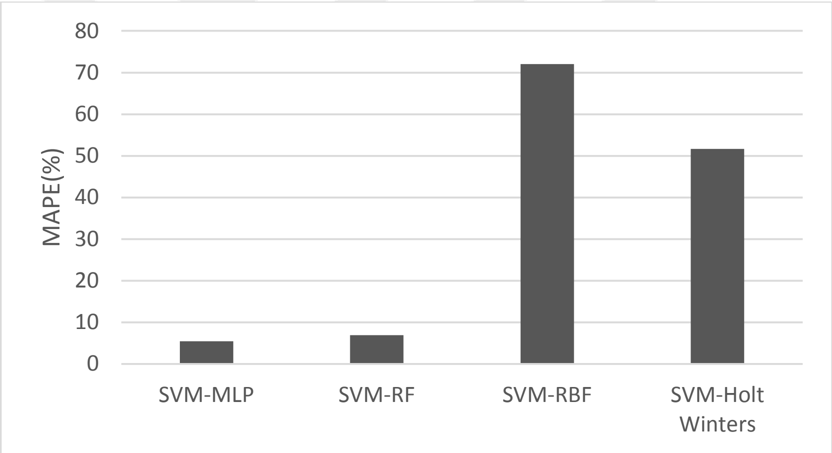

Figure 5.22. Percentage decrease rates in average *MAPE* of the forecasting models for hourly PS downlink traffic with *90%-10%* split rate

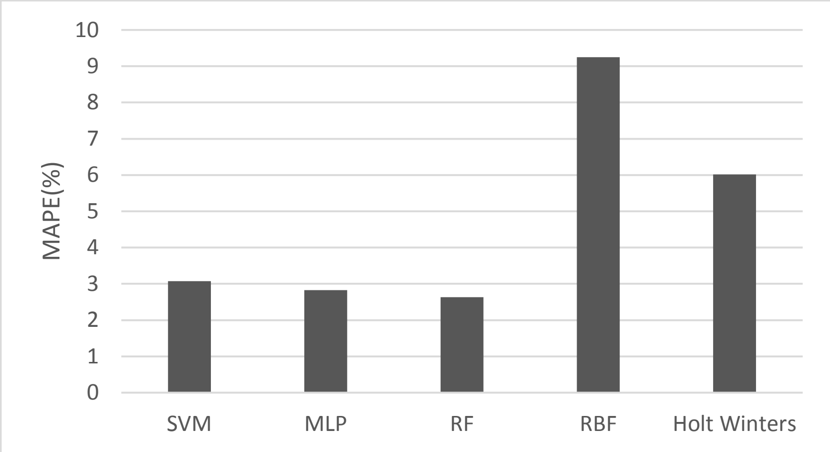

Figure 5.23. Average *MAPE* of the forecasting models for hourly PS uplink traffic with *90%-10%* split rate

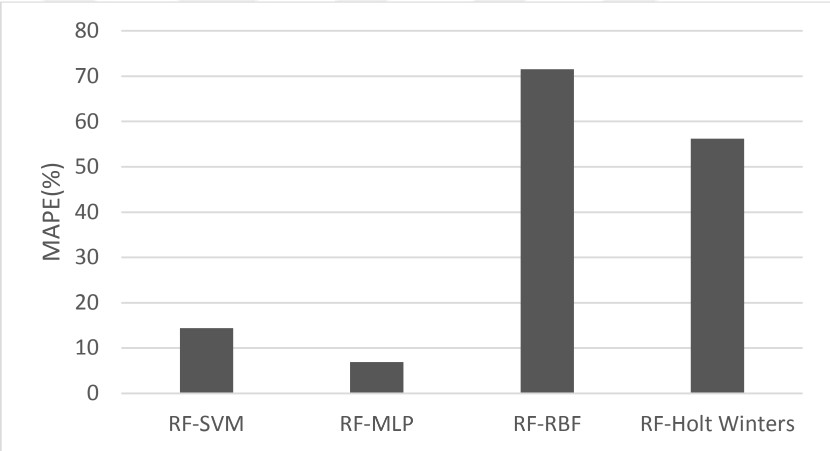

Figure 5.24. Percentage decrease rates in average *MAPE* of the forecasting models for hourly PS uplink traffic with *90%-10%* split rate

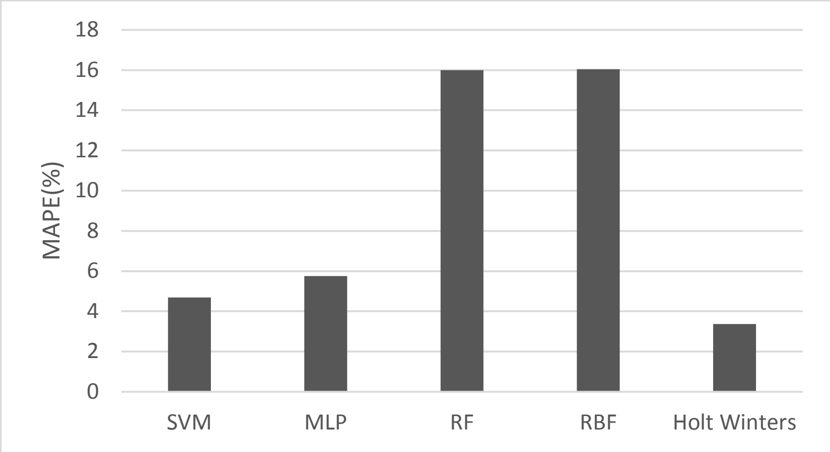

Figure 5.25. Average *MAPE* of the forecasting models for daily CS voice traffic with *70%-30%* split rate

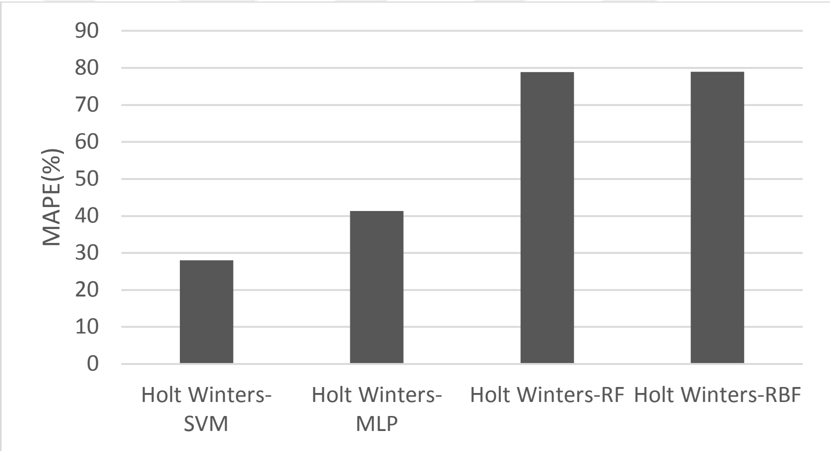

Figure 5.26. Percentage decrease rates in average *MAPE* of the forecasting models for daily CS voice traffic with *70%-30%* split rate

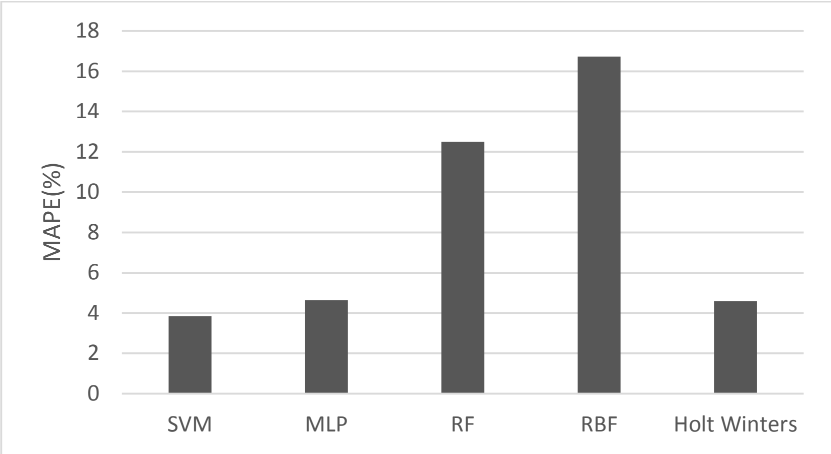

Figure 5.27. Average *MAPE* of the forecasting models for daily PS total traffic with *70%-30%* split rate

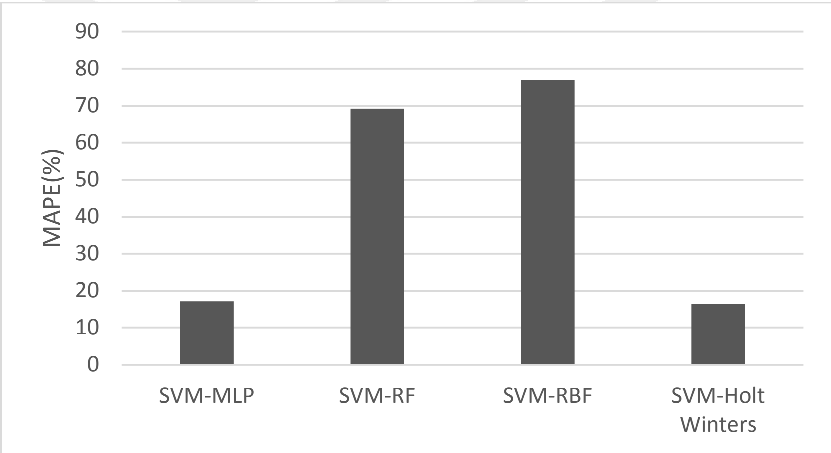

Figure 5.28. Percentage decrease rates in average *MAPE* of the forecasting models for daily PS total traffic with *70%-30%* split rate

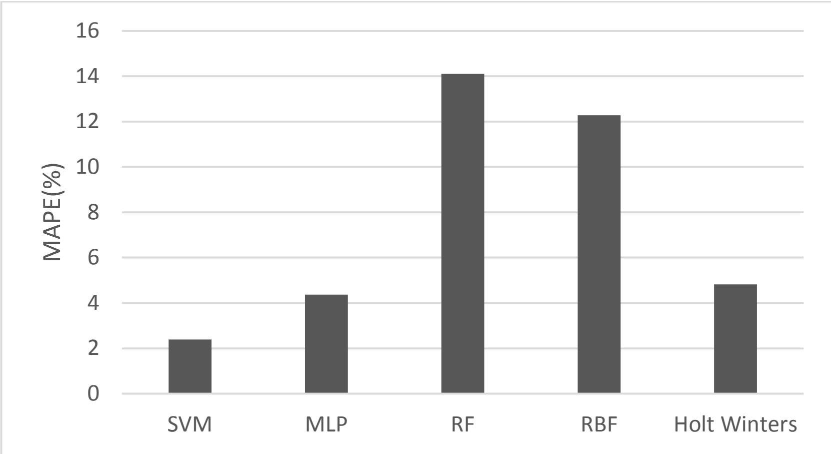

Figure 5.29. Average *MAPE* of the forecasting models for daily PS downlink traffic with *70%-30%* split rate

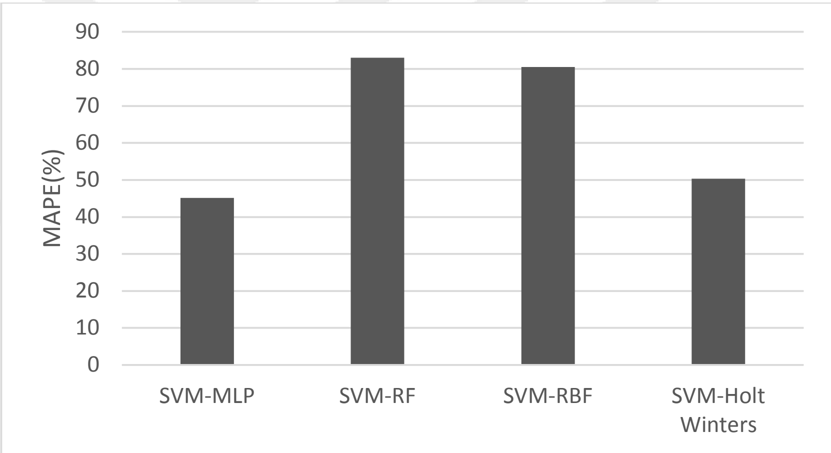

Figure 5.30. Percentage decrease rates in average *MAPE* of the forecasting models for daily PS downlink traffic with *70%-30%* split rate

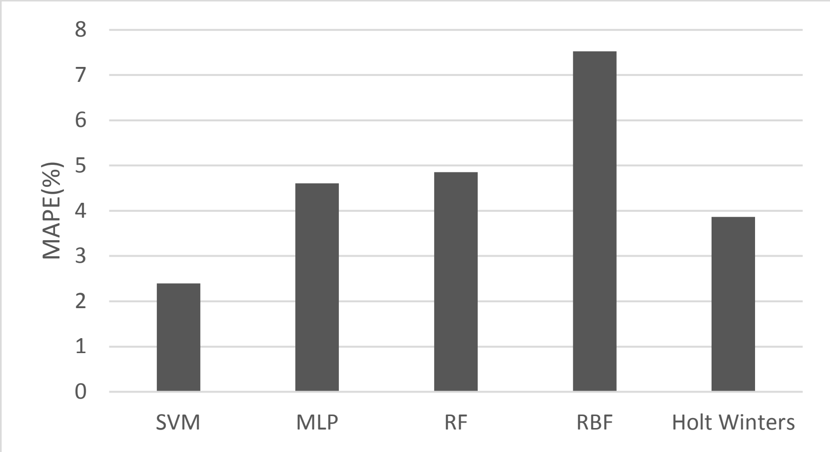

Figure 5.31. Average *MAPE* of the forecasting models for daily PS uplink traffic with *70%-30%* split rate

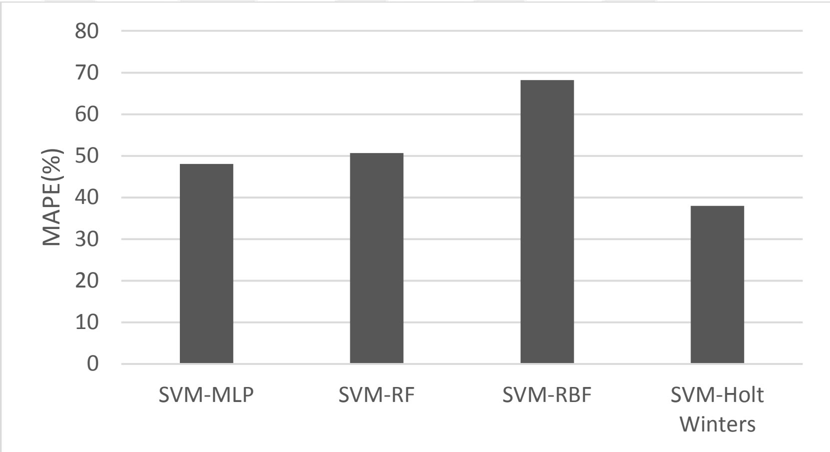

Figure 5.32. Percentage decrease rates in average *MAPE* of the forecasting models for daily PS uplink traffic with *70%-30%* split rate

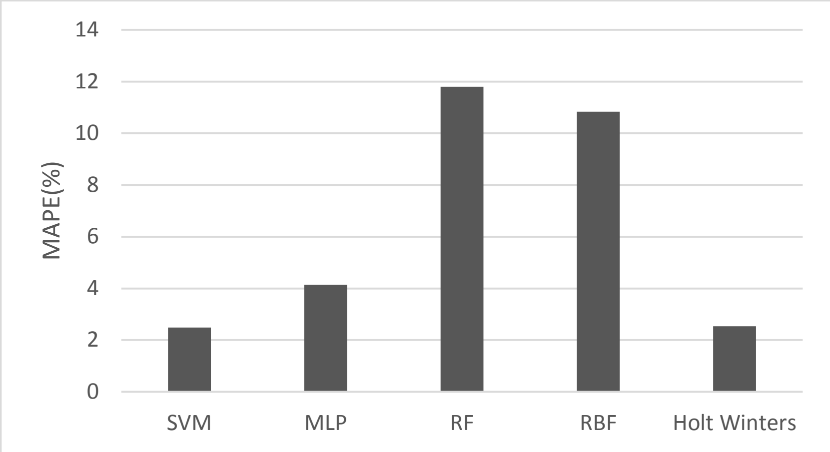

Figure 5.33. Average *MAPE* of the forecasting models for daily CS voice traffic with *80%-20%* split rate

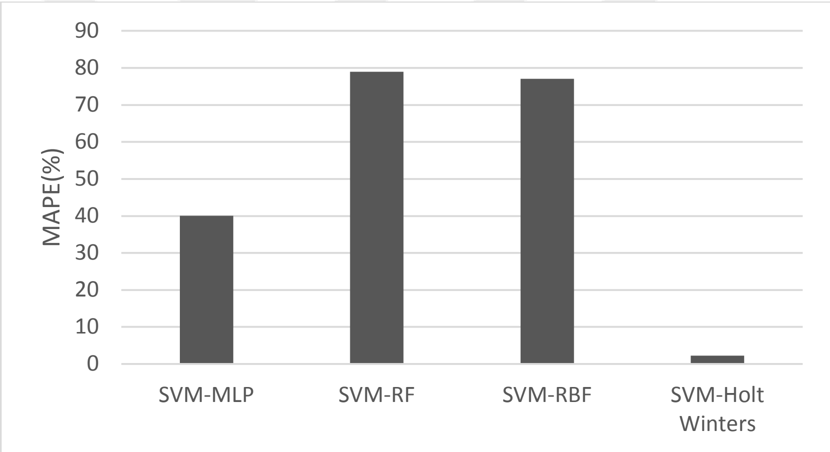

Figure 5.34. Percentage decrease rates in average *MAPE* of the forecasting models for daily CS voice traffic with *80%-20%* split rate

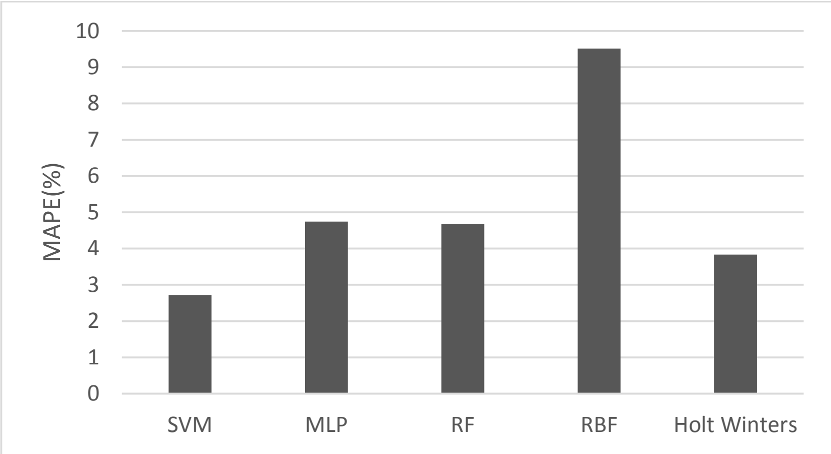

Figure 5.35. Average *MAPE* of the forecasting models for daily PS total traffic with *80%-20%* split rate

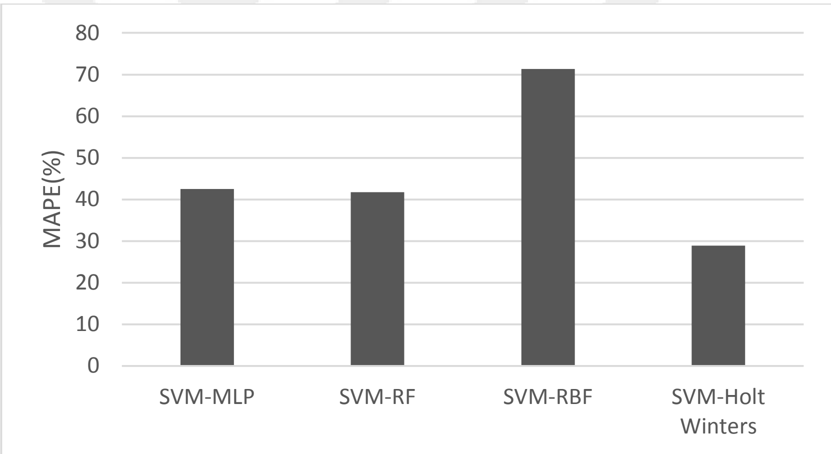

Figure 5.36. Percentage decrease rates in average *MAPE* of the forecasting models for daily PS total traffic with *80%-20%* split rate

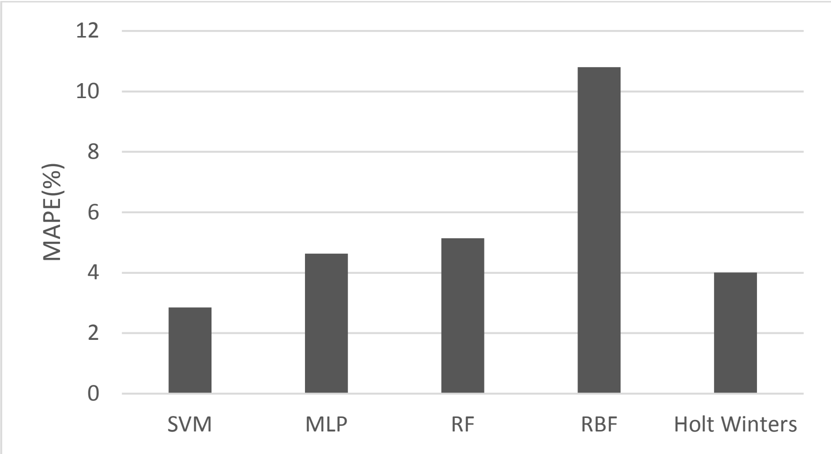

Figure 5.37. Average *MAPE* of the forecasting models for daily PS downlink traffic with *80%-20%* split rate

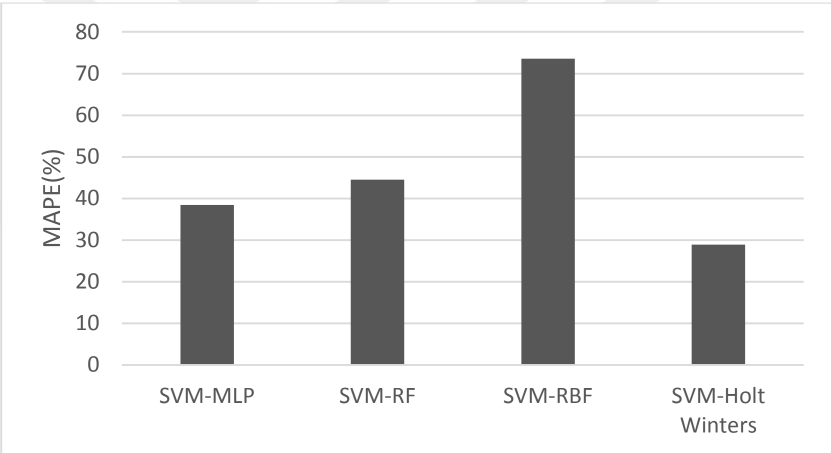

Figure 5.38. Percentage decrease rates in average *MAPE* of the forecasting models for daily PS downlink traffic with *80%-20%* split rate

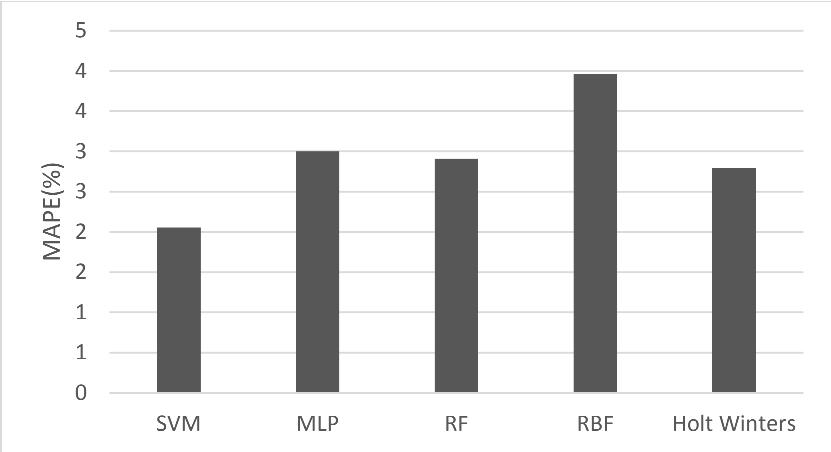

Figure 5.39. Average *MAPE* of the forecasting models for daily PS uplink traffic with *80%-20%* split rate

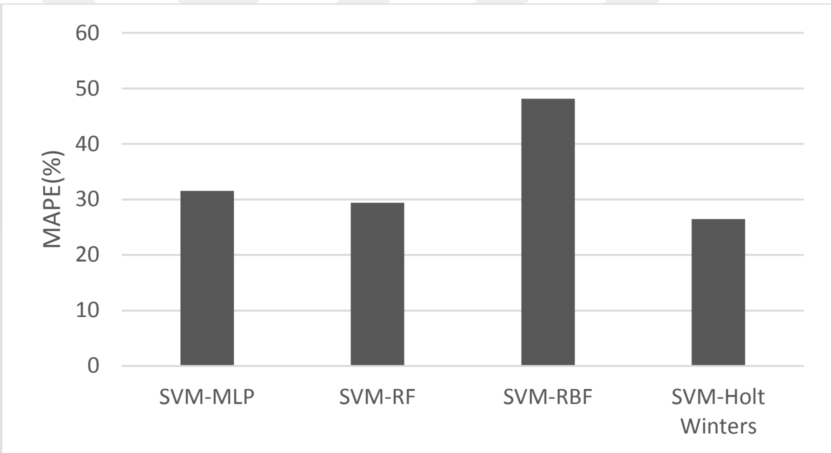

Figure 5.40. Percentage decrease rates in average *MAPE* of the forecasting models for daily PS uplink traffic with *80%-20%* split rate

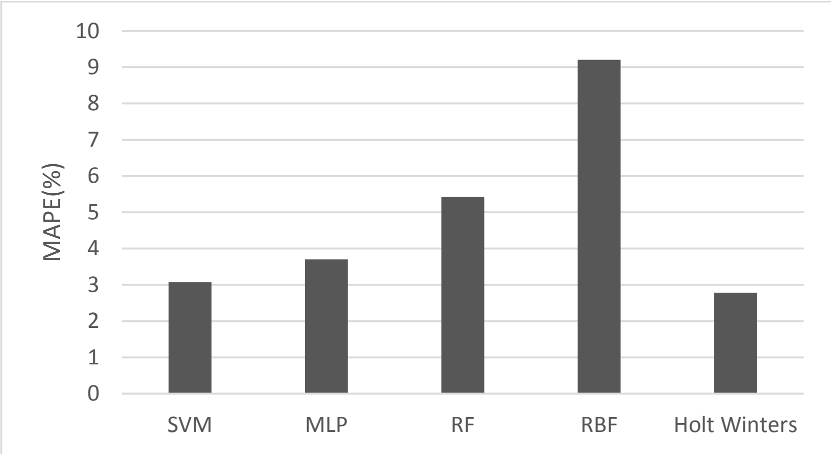

Figure 5.41. Average *MAPE* of the forecasting models for daily CS voice traffic with *90%-10%* split rate

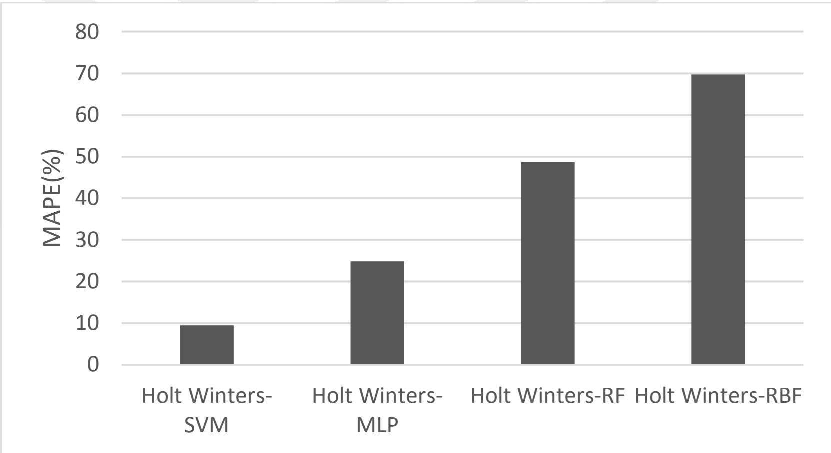

Figure 5.42. Percentage decrease rates in average *MAPE* of the forecasting models for daily CS voice traffic with *90%-10%* split rate

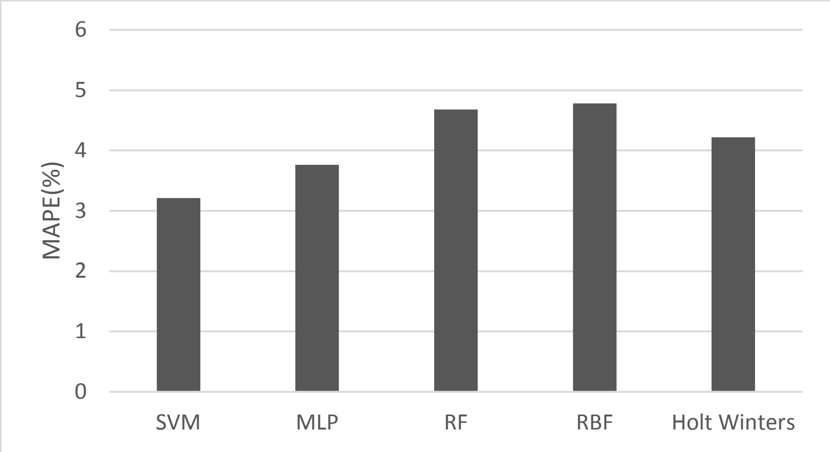

Figure 5.43. Average *MAPE* of the forecasting models for daily PS total traffic with *90%-10%* split rate

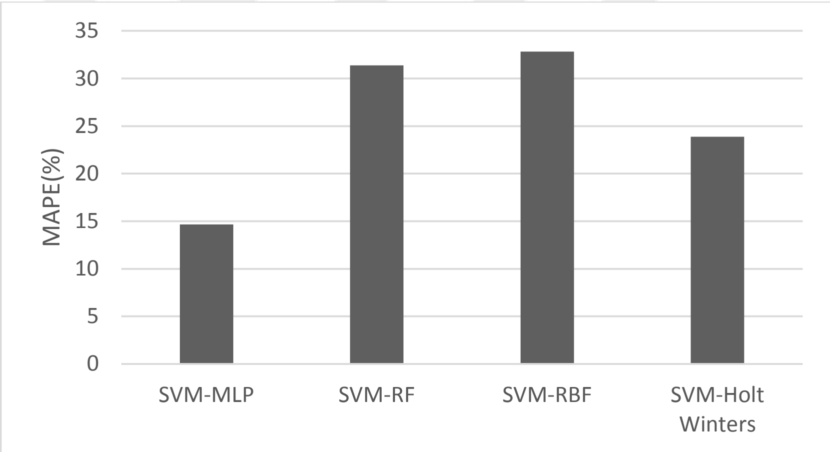

Figure 5.44. Percentage decrease rates in average *MAPE* of the forecasting models for daily PS total traffic with *90%-10%* split rate

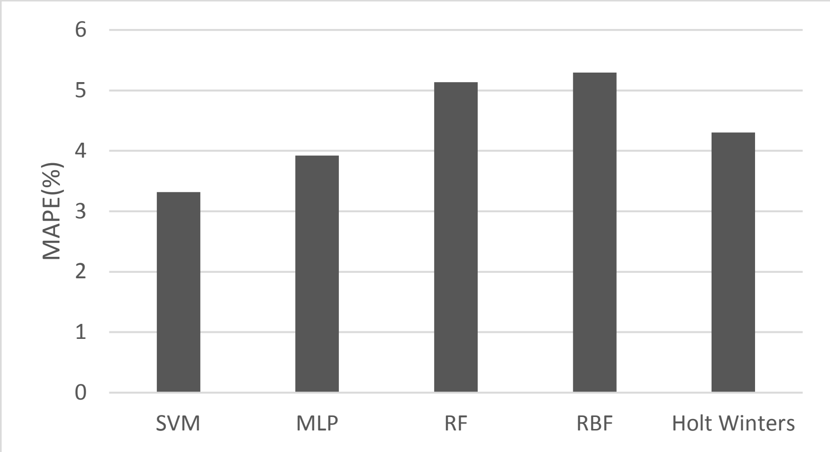

Figure 5.45. Average *MAPE* of the forecasting models for daily PS downlink traffic with *90%-10%* split rate

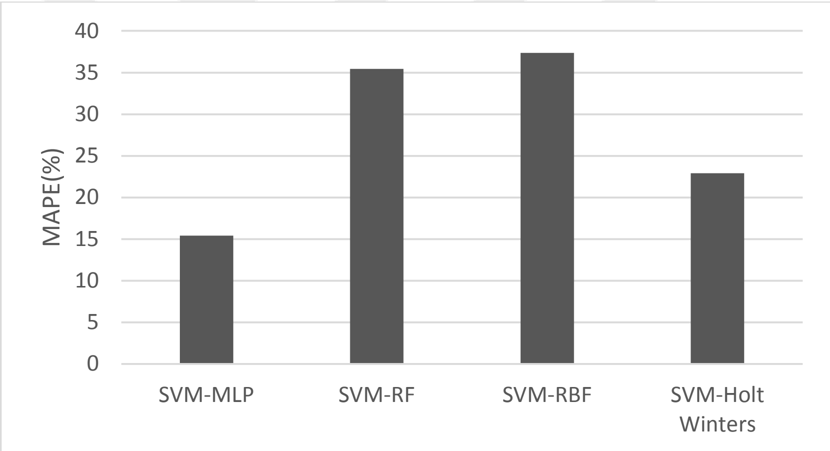

Figure 5.46. Percentage decrease rates in average *MAPE* of the forecasting models for daily PS downlink traffic with *90%-10%* split rate

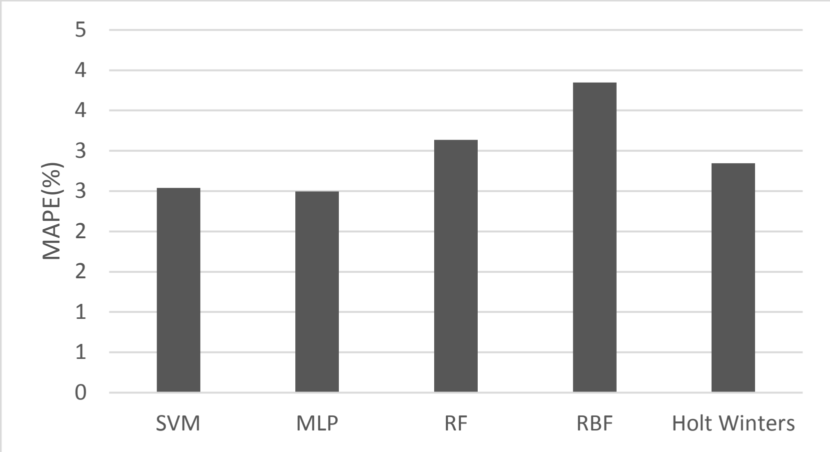

Figure 5.47. Average *MAPE* of the forecasting models for daily PS uplink traffic with *90%-10%* split rate

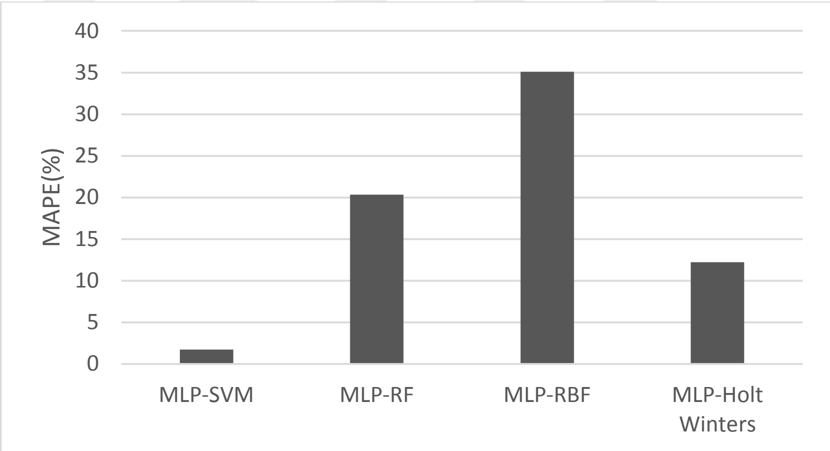

Figure 5.48. Percentage decrease rates in average *MAPE* of the forecasting models for daily PS uplink traffic with *90%-10%* split rate

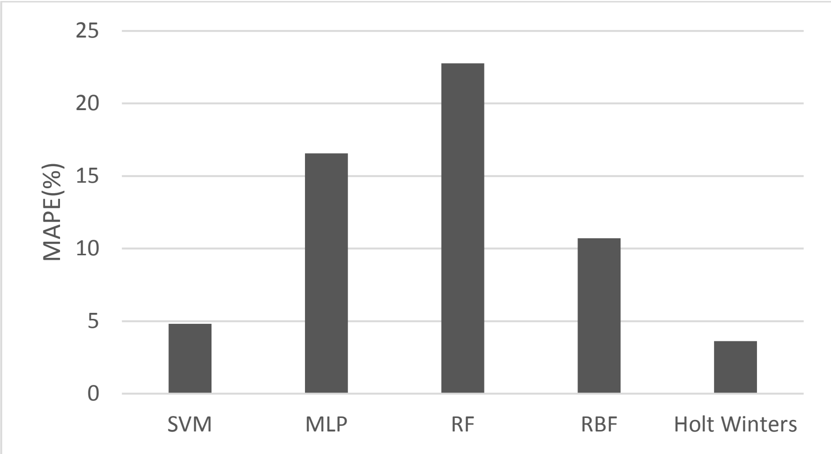

Figure 5.49. Average *MAPE* of the forecasting models for weekly CS voice traffic with *70%-30%* split rate

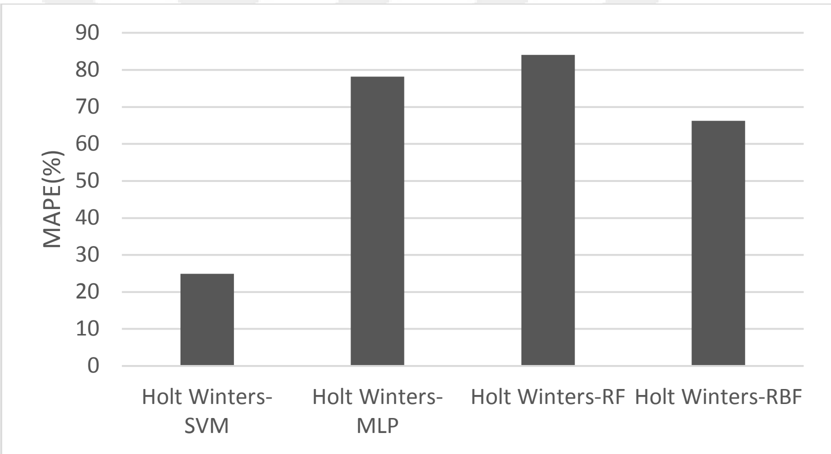

Figure 5.50. Percentage decrease rates in average *MAPE* of the forecasting models for weekly CS voice traffic with *70%-30%* split rate

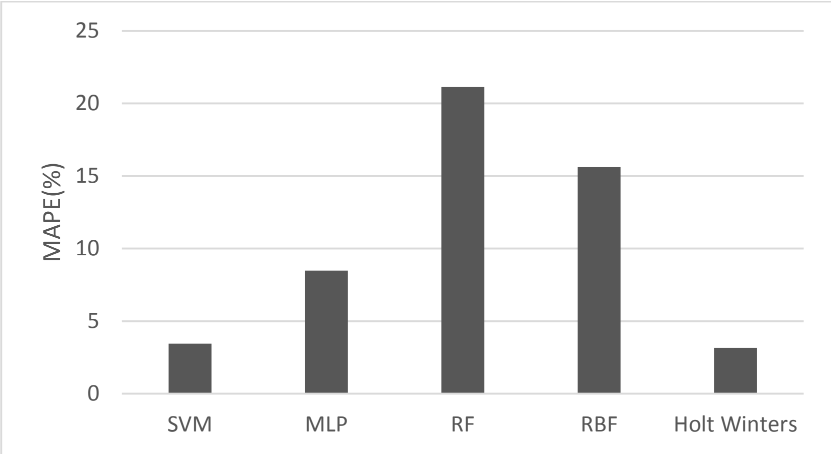

Figure 5.51. Average *MAPE* of the forecasting models for weekly PS total traffic with *70%-30%* split rate

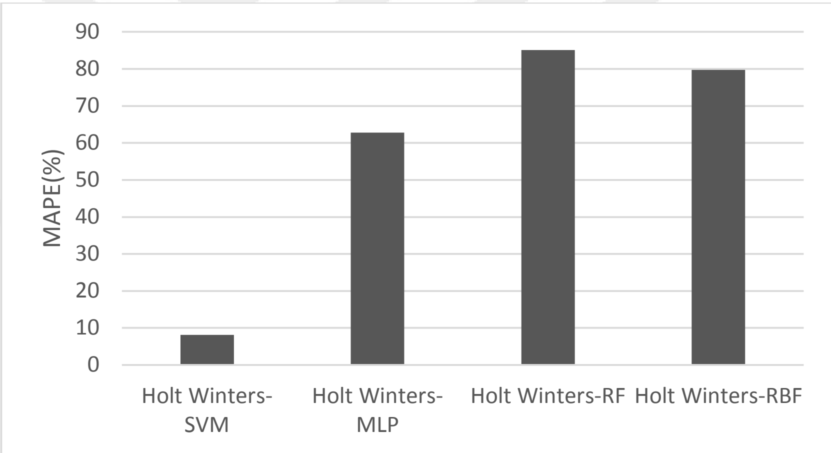

Figure 5.52. Percentage decrease rates in average *MAPE* of the forecasting models for weekly PS total traffic with *70%-30%* split rate

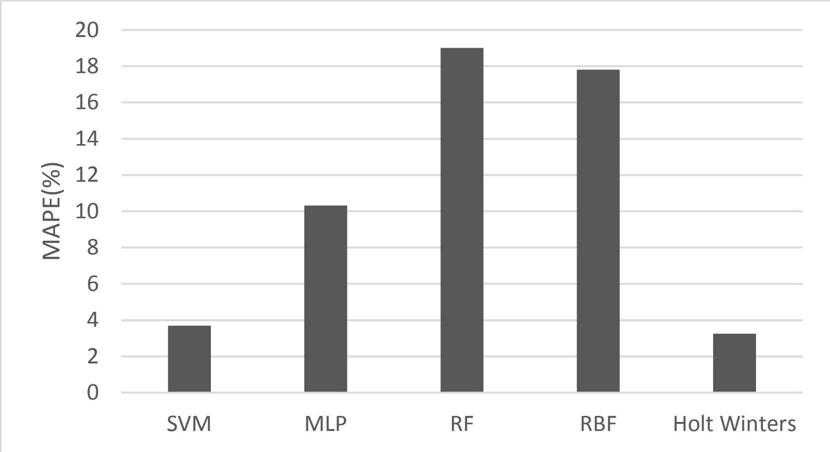

Figure 5.53. Average *MAPE* of the forecasting models for weekly PS downlink traffic with *70%-30%* split rate

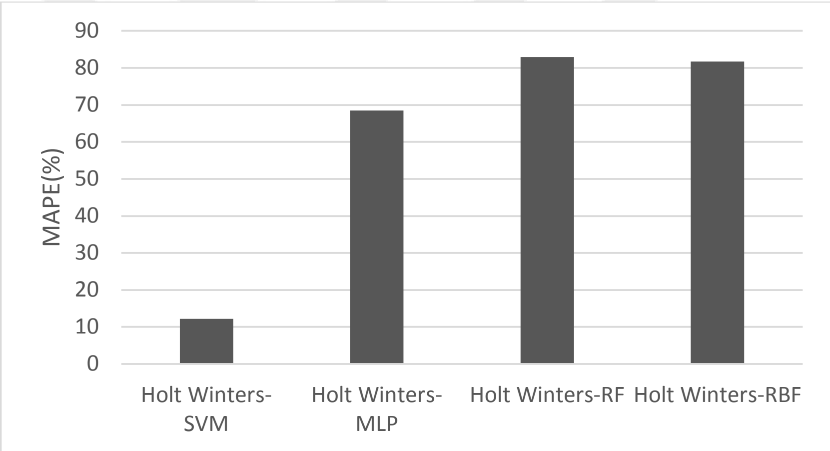

Figure 5.54. Percentage decrease rates in average *MAPE* of the forecasting models for weekly PS downlink traffic with *70%-30%* split rate

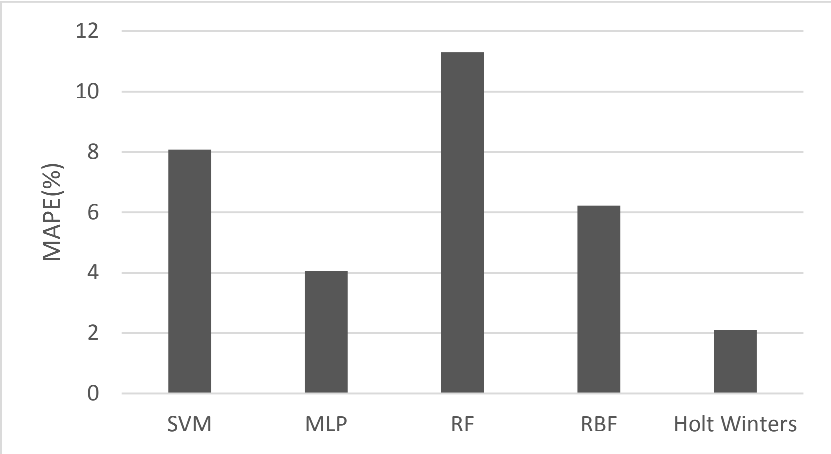

Figure 5.55. Average *MAPE* of the forecasting models for weekly PS uplink traffic with *70%-30%* split rate

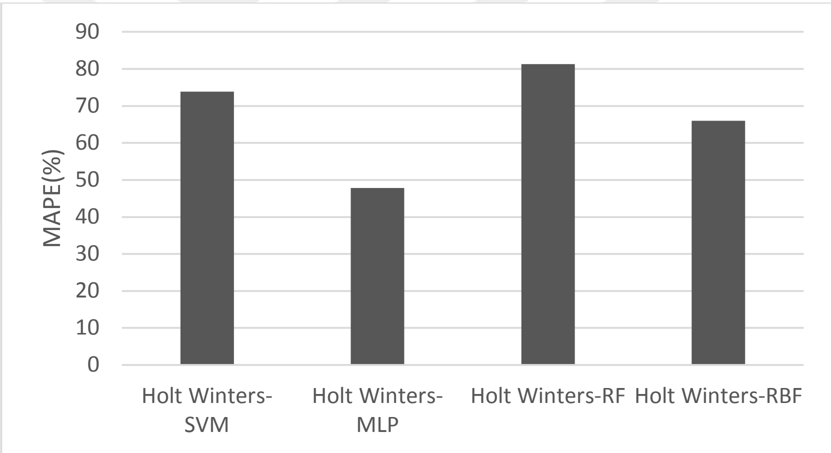

Figure 5.56. Percentage decrease rates in average *MAPE* of the forecasting models for weekly PS uplink traffic with *70%-30%* split rate

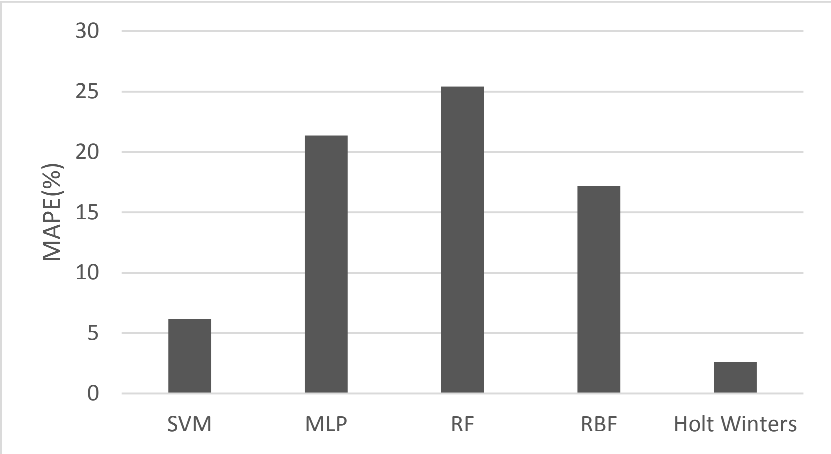

Figure 5.57. Average *MAPE* of the forecasting models for weekly CS voice traffic with *80%-20%* split rate

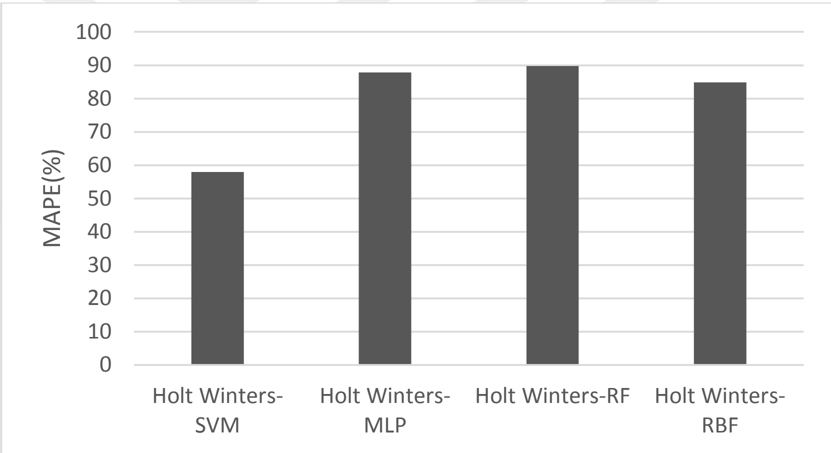

Figure 5.58. Percentage decrease rates in average *MAPE* of the forecasting models for weekly CS voice traffic with *80%-20%* split rate

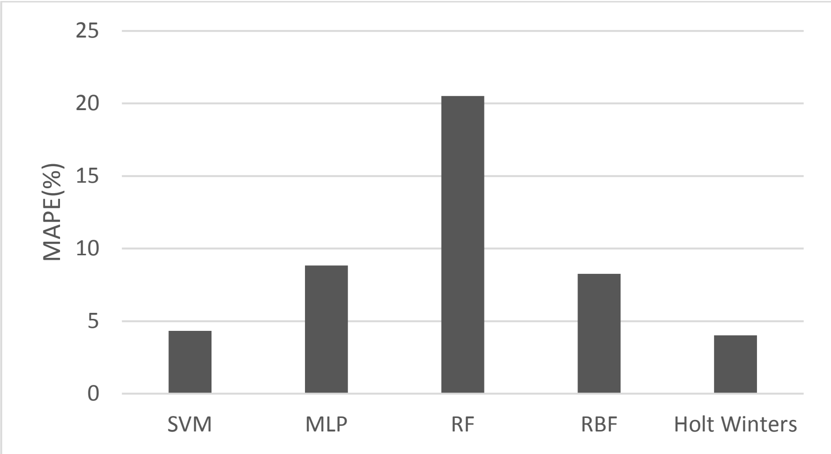

Figure 5.59. Average *MAPE* of the forecasting models for weekly PS total traffic with *80%-20%* split rate

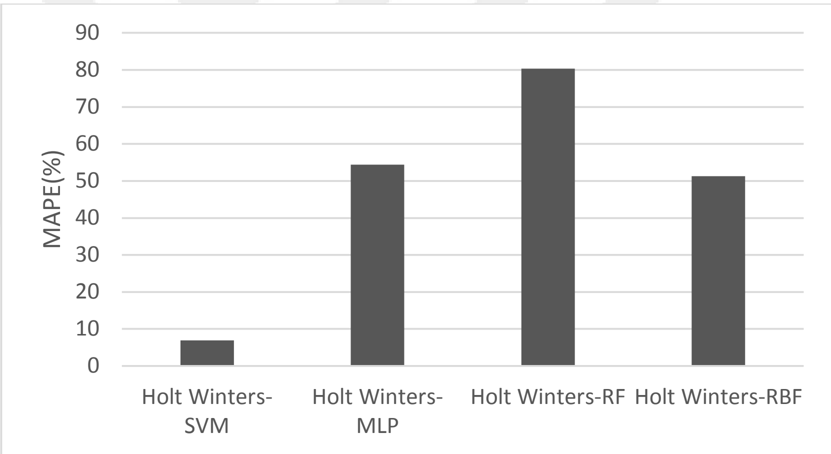

Figure 5.60. Percentage decrease rates in average *MAPE* of the forecasting models for weekly PS total traffic with *80%-20%* split rate

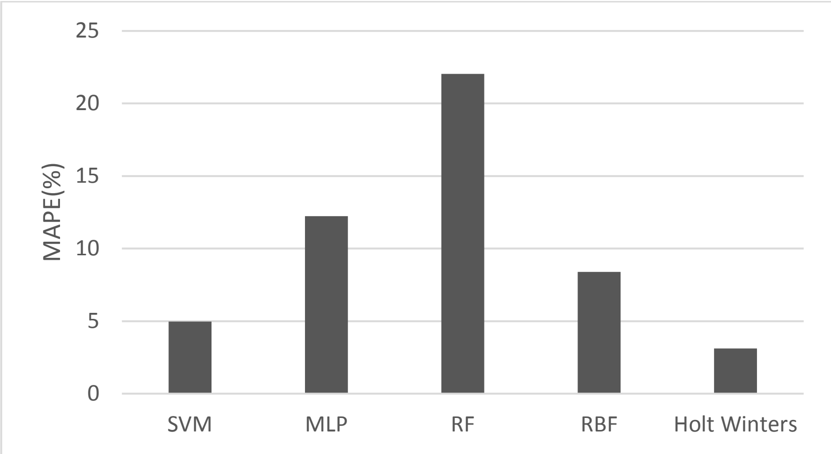

Figure 5.61. Average *MAPE* of the forecasting models for weekly PS downlink traffic with *80%-20%* split rate

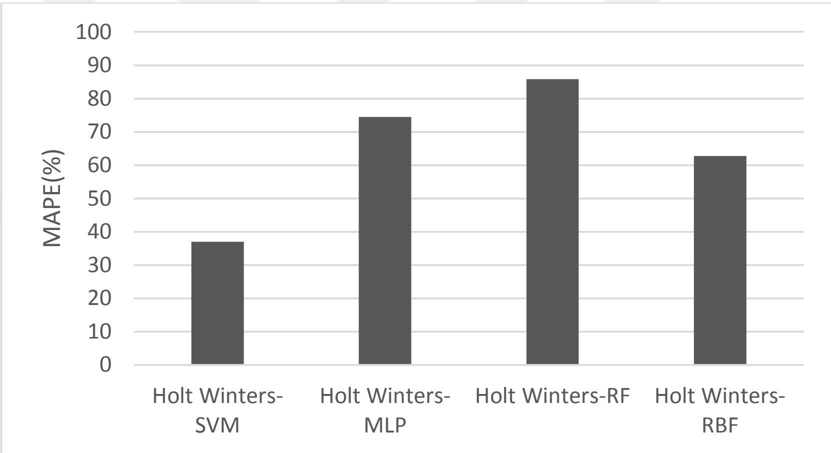

Figure 5.62. Percentage decrease rates in average *MAPE* of the forecasting models for weekly PS downlink traffic with *80%-20%* split rate

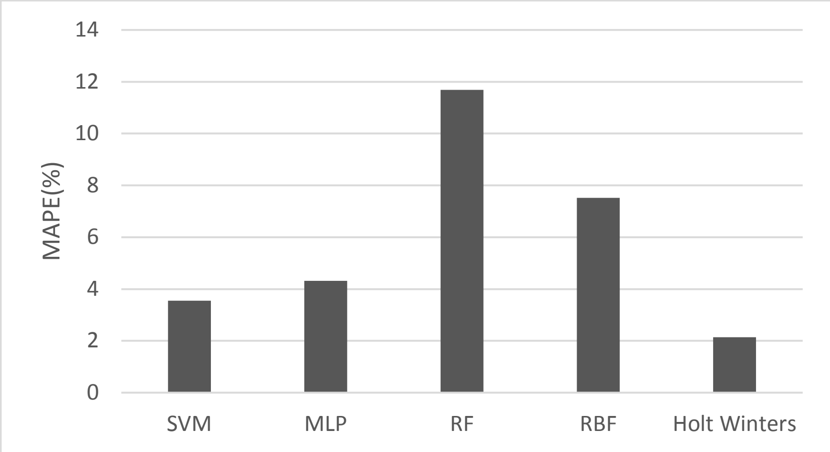

Figure 5.63. Average *MAPE* of the forecasting models for weekly PS uplink traffic with *80%-20%* split rate

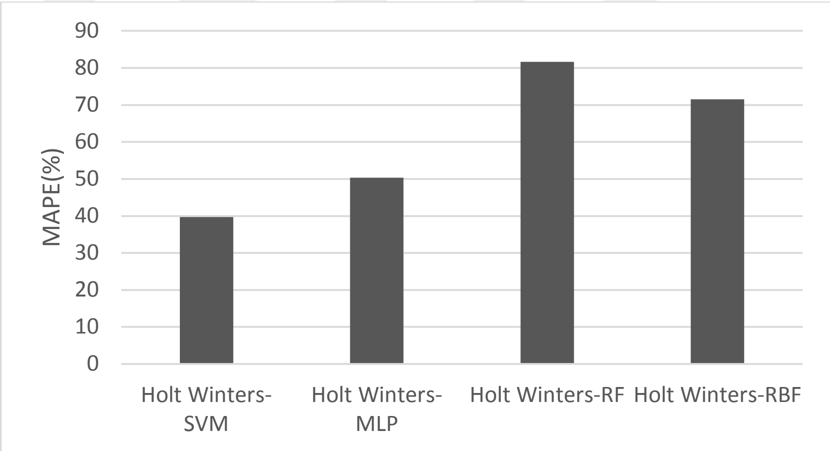

Figure 5.64. Percentage decrease rates in average *MAPE* of the forecasting models for weekly PS uplink traffic with *80%-20%* split rate

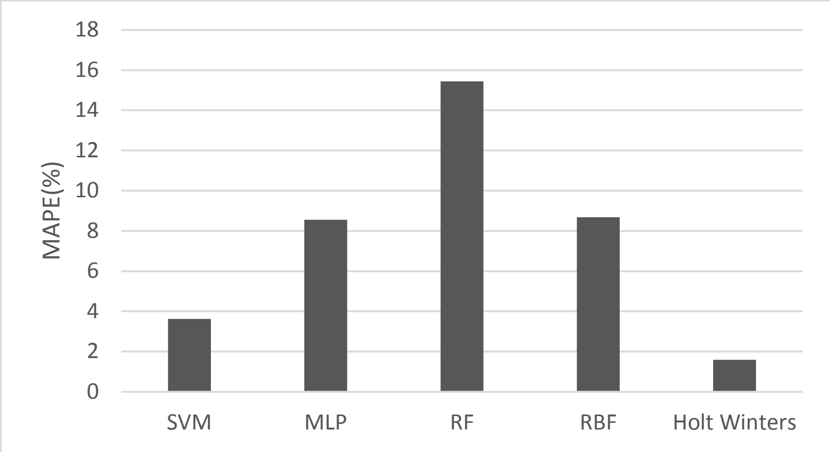

Figure 5.65. Average *MAPE* of the forecasting models for weekly CS voice traffic with *90%-10%* split rate

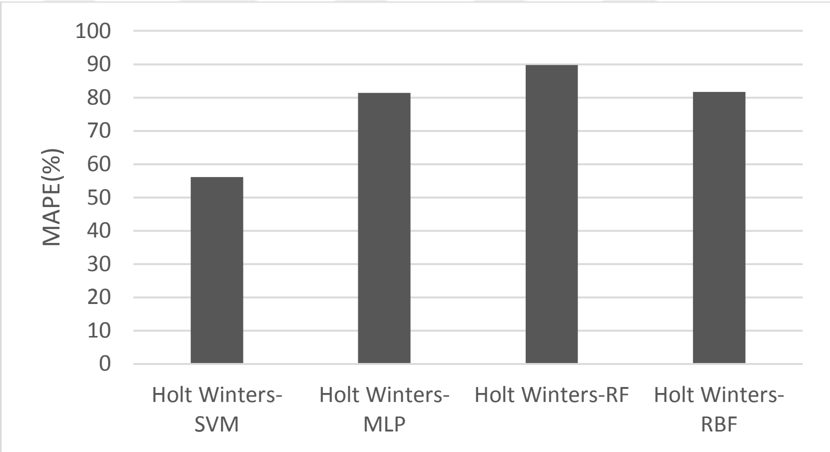

Figure 5.66. Percentage decrease rates in average *MAPE* of the forecasting models for weekly CS voice traffic with *90%-10%* split rate

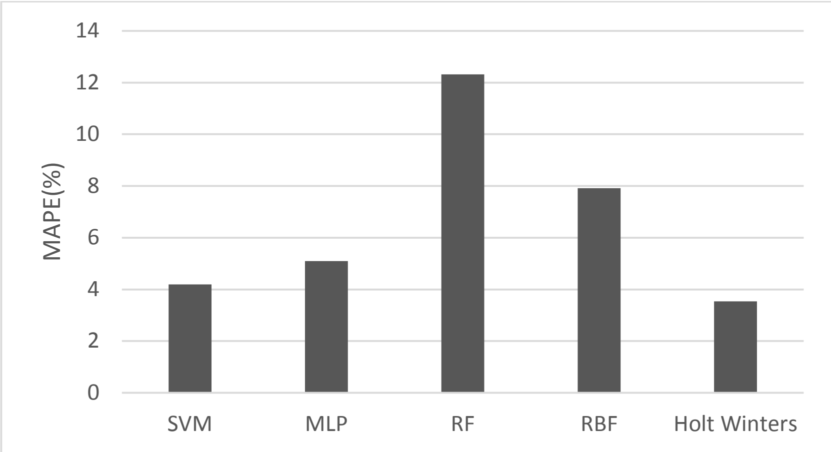

Figure 5.67. Average *MAPE* of the forecasting models for weekly PS total traffic with *90%-10%* split rate

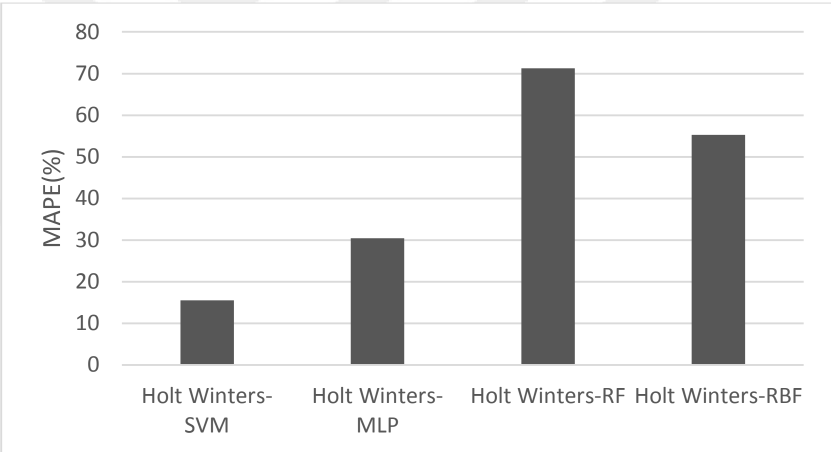

Figure 5.68. Percentage decrease rates in average *MAPE* of the forecasting models for weekly PS total traffic with *90%-10%* split rate

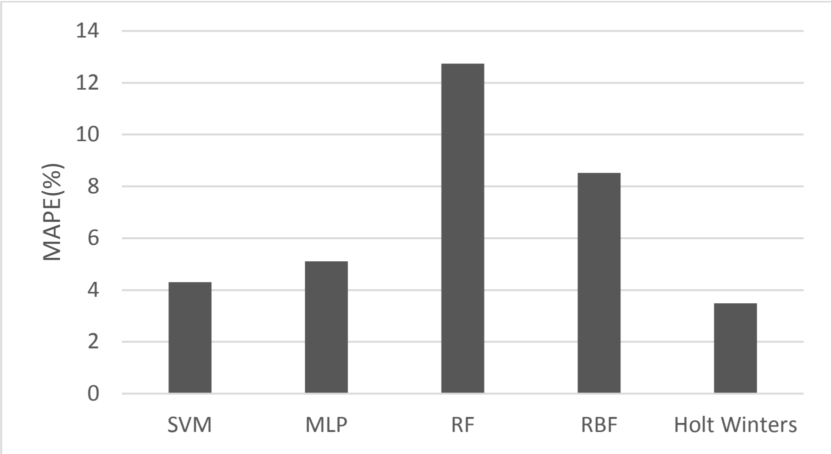

Figure 5.69. Average *MAPE* of the forecasting models for weekly PS downlink traffic with *90%-10%* split rate

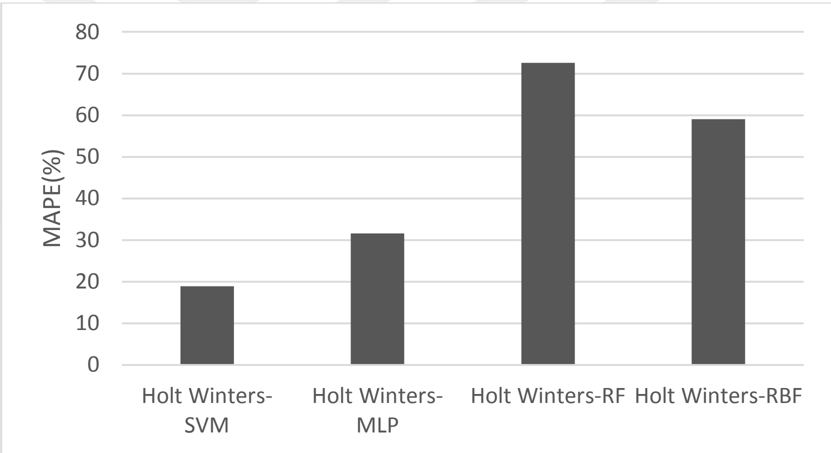

Figure 5.70. Percentage decrease rates in average *MAPE* of the forecasting models for weekly PS downlink traffic with *90%-10%* split rate

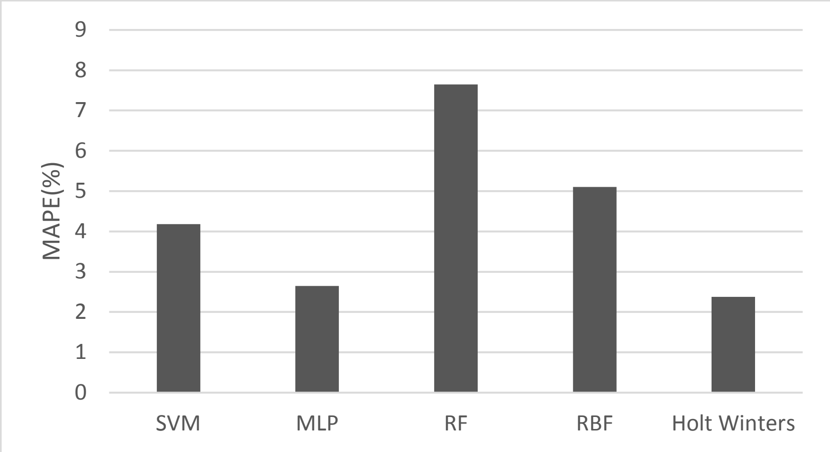

Figure 5.71. Average *MAPE* of the forecasting models for weekly PS uplink traffic with *90%-10%* split rate

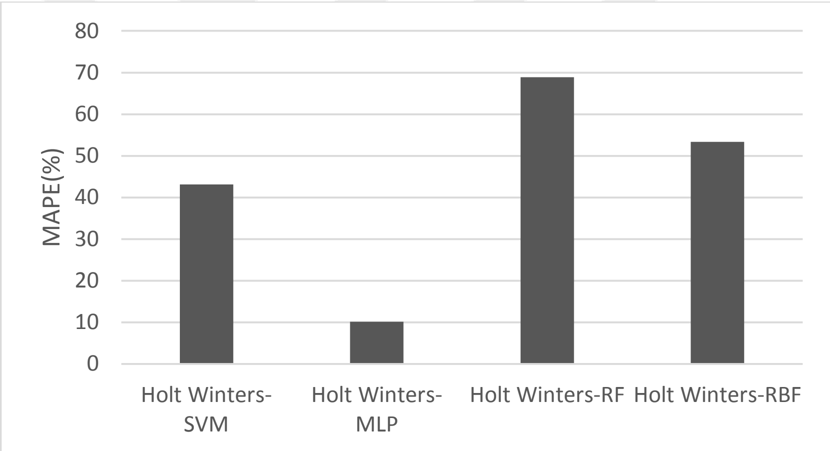

Figure 5.72. Percentage decrease rates in average *MAPE* of the forecasting models for weekly PS uplink traffic with *90%-10%* split rate

5.1. General Discussion on the Results

- For all data sets, in general, SVM and Holt-Winters based prediction models show the highest performance regardless of the selection of time lags. Among the regression models, the general ranking of the methods in terms of their prediction performance based on the *MAPE*'s is SVM, Holt-Winters, MLP, RF and RBF.

- Generally, the results show that prediction models based on the daily time scale exhibit higher performance than the models based on the other time scales (hourly and weekly). But there is no distinct performance difference between the prediction models based on other time scales (hourly and weekly).
- For hourly and daily datasets, in general, the time lags having shorter time scale (Lag-2) yield higher *MAPE's*. On the other hand, shorter time lag outperforms for the weekly dataset.
- For hourly dataset, the time lags having longer time scale (Lag-4) yields best *MAPE* results for all split ranges and for all type of traffic datasets. However, there are no distinct performance differences for other datasets (daily and weekly) and split ranges.
- When the performance of the prediction models based on averages *MAPE's* combined with all split ranges is examined, the *MAPE's* of the prediction models obtained on daily dataset have lower error rates than the ones obtained on hourly ad weekly time scale.
- Lowest *MAPE* values are achieved on weekly CS voice dataset via 90% to 10% train-test split range. On the other hand, the highest MAPE values are acquired on hourly CS voice dataset with a split range of 70%_30%.
- Network dimensioning and future capacity enhancement can be conducted by using the predicted traffic and its time granularity. The expansion of the network parts (RAN or Core) can be easily controlled by using most relevant traffic type which limits the capacity of the system frequently.

5.2. Discussion on Hourly Dataset Results

- For hourly traffic datasets, in general, SVM-based prediction models have lower *MAPE* values than the prediction models based on other machine learning and statistical methods regardless of the time lags selection with a single exception.
- For all hourly traffic datasets, in general, RBF yields the worst prediction performance with respect to the highest average *MAPE's.*
- In general, the interval of the *MAPE's* of the prediction models for hourly datasets are between 3.31% to 94.82% for CS voice traffic, 2.10% to 19.62% for PS total traffic, 2.06% to 19.23% for PS downlink traffic and 1.95% to 23.56% for PS uplink traffic.
- For CS voice traffic, the general ranking of the models in terms of their prediction performance based on the average *MAPE's* is SVM, MLP, RF, Holt-Winters and RBF for 70%_30% split; SVM, RF, MLP, Holt-Winters and RBF for 80%_20% split and SVM, RF, Holt-Winters, MLP and RBF for 90%_10% split.
- For PS total traffic, the general ranking of the models in terms of their prediction performance based on the average *MAPE's* is SVM, RF, MLP, Holt-Winters and RBF for 70%_30% split; RF, SVM, MLP, Holt-Winters and RBF for 80%_20% split and SVM, MLP, RF, Holt-Winters, MLP and RBF for 90%_10% split.
- For PS downlink traffic, the general ranking of the models in terms of their prediction performance based on the average *MAPE's* is SVM, RF, MLP, Holt-Winters and RBF for 70%_30% split; RF, SVM, MLP, Holt-Winters and RBF for 80%_20% split and SVM, MLP, MLP, RF, Holt-Winters and RBF for 90%_10% split.

- For PS uplink traffic, the general ranking of the models in terms of their prediction performance based on the average *MAPE's* is RF, SVM, MLP, Holt-Winters and RBF for 70%_30% split; MLP, SVM, RF, Holt-Winters and RBF for 80%_20% split and RF, MLP, SVM, Holt-Winters and RBF for 90%_10% split.
- Among the lengths of time lags, the lags having the longer scale or generated by selecting all autocorrelations above a given threshold give lower *MAPE*'s for all hourly traffic datasets. Particularly, Lag-4 yield 22.96%, 50.7%, 30% lower *MAPE*'s on the average than the *MAPE*'s of Lag-1, Lag-2 and Lag-3 for voice traffic, 10.99%, 22.93% and 10.18% for PS total traffic, 13.17%, 30.43% and 17.11% for PS downlink traffic, 5.05%, 25.16% and 11.12% for PS uplink traffic.
- In general, the lowest *MAPE* values and the lowest average *MAPE's* are acquired via 90%_10% split for all hourly traffic datasets. More precisely, it yields 30.09% and 19.12% lower *MAPE's* on the average than the *MAPE's* of 70%_30% and 80%_20% splits.

5.3. Discussion on Daily Dataset Results

- For daily traffic datasets based on 80%_20% split, SVM-based prediction models have lower *MAPE* values than the prediction models based on other machine learning and statistical methods regardless of the time lags selection. For other splits of the dataset, general ranking among the methods in terms of best result is SVM, Holt-Winters, and MLP.

- For all daily traffic datasets, in general, RBF shows the worst performance based on the prediction performance with the highest average *MAPE's.*
- In general, the interval of the *MAPE's* of the prediction models for daily datasets is between 2.25% to 18.93% for CS voice traffic, 2.89% to 18.49% for PS total traffic, 2.18% to 14.84% for PS downlink traffic and 2.02% to 8.73% for PS uplink traffic.
- For CS voice traffic, the general ranking of the models in terms of their prediction performance based on the average *MAPE's* is Holt-Winters, SVM, MLP, RF and RBF for 70%_30% split; SVM, Holt-Winters, MLP, RBF and RF for 80%_20% split and Holt-Winters, SVM, MLP, RF and RBF for 90%_10% split range.
- For PS total traffic, the general ranking of the models in terms of their prediction performance based on the average *MAPE's* is SVM, Holt-Winters, MLP, RF and RBF for 70%_30% split; SVM, Holt-Winters, RF, MLP and RBF for 80%_20% split and SVM, MLP, Holt-Winters, MLP, RF and RBF for 90%_10% split.
- For PS downlink traffic, the general ranking of the models in terms of their prediction performance based on the average *MAPE's* is SVM, MLP, Holt-Winters, RBF and RF for 70%_30% split; SVM, Holt-Winters, MLP, RF and RBF for 80%_20% split and SVM, MLP, Holt-Winters, RF and RBF for 90%_10% split.
- For PS uplink traffic, the general ranking of the models in terms of their prediction performance based on the average *MAPE's* is SVM, Holt-Winters, MLP, RF and RBF for 70%_30% split; SVM, Holt-Winters, RF, MLP and RBF for 80%_20% split and MLP, SVM, Holt-Winters, RF and RBF for 90%_10% split.

- In general, the lowest *MAPE* values and the lowest average *MAPE's* are acquired via 80%_20% split for all hourly traffic datasets with a single exception. It yields 14.39% and 15.83% lower *MAPE's* on the average than the *MAPE's* of 90%_10% and 70%_30% splits.
- Among the lengths of time lags, the lags having the longer scale or generated by selecting all autocorrelations above a given threshold give lower *MAPE*'s only for PS total and PS uplink traffic. Particularly, Lag-4 yield 12.12%, 4.91% lower *MAPE*'s on the average than the *MAPE*'s of Lag-1 and Lag-3 for PS total traffic, 8.03%, 4.39% and 6.11% lower than Lag-1, Lag-2, and Lag-3 for PS downlink traffic.

5.4. Discussion on Weekly Dataset Results

- For weekly traffic datasets, Holt-Winters based prediction models have lower average *MAPE* values than the prediction models based on the machine learning methods regardless of the time lags and split range selection. More specifically, Holt-Winters prediction models yield 46.64%, 83.23%, 87.74% and 78.68% lower *MAPE's* on the average than the *MAPE's* of SVM, MLP, RF, RBF for CS voice traffic; 10.29%, 52.13%, 80.13%, 66.28% for PS total traffic; %23.74, 64.24%, 81.62%, 71.54% for PS downlink traffic; %58.06%, 39.73%, 78.35% and 64.81% for PS uplink traffic, respectively.
- For all weekly traffic datasets, in general, RF shows the worst performance based on the highest average *MAPE's.*
- In general, the interval of the *MAPE's* of the prediction models for weekly traffic datasets are between 1.59% to 26.32% for CS voice

traffic, 3.12% to 23.38% for PS total traffic, 2.18% to 14.84% for PS downlink traffic and 2.01% to 13.48% for PS uplink traffic.

- For CS voice traffic, the general ranking of the models in terms of their prediction performance based on the average *MAPE's* is Holt-Winters, SVM, RBF, MLP and RF for 70%_30% split; Holt-Winters, SVM, RBF, MLP and RF for 80%_20% split and Holt-Winters, SVM, MLP, RBF and RF for 90%_10% split.
- For PS total traffic, the general ranking of the models in terms of their prediction performance based on the average *MAPE's* is Holt-Winters, SVM, RBF and RF for 70%_30% split; Holt-Winters, SVM, RBF, MLP and RF for 80%_20% split and Holt-Winters, SVM, MLP, RBF and RF for 90%_10% split.
- For PS downlink traffic, the general ranking of the models in terms of their prediction performance based on the average *MAPE's* is Holt-Winters, SVM, MLP, RBF and RF for 70%_30% split; Holt-Winters, SVM, RBF, MLP and RF for 80%_20% split and Holt-Winters, SVM, MLP, RBF and RF for 90%_10% split.
- For PS uplink traffic, the general ranking of the models in terms of their prediction performance based on the average *MAPE's* is Holt-Winters, SVM, MLP, RBF and RF for 70%_30% split; Holt-Winters, SVM, RBF, MLP and RF for 80%_20% split and Holt-Winters, SVM, MLP, RBF and RF for 90%_10% split.

6. CONCLUSION

In this thesis, circuit switched voice and packet switched data traffic prediction models have been developed for commercially deployed 3G/UMTS network in Turkey using various machine learning methods including SVM, MLP, RBF, RF and a statistical regression method which is Holt-Winters. Experiments have been conducted on twelve different data sets which have been formed by different time scales and carried traffic type. Several time lags have been utilized for each data set to develop voice and data traffic forecasting models. For model training and testing, the utilized dataset has been partitioned in three different ranges. In the first split, 70% of dataset content has been used for training while the rest of data has been used for testing. Additionally, 80%_20% and 90%_10% train and test splits have been used as second and third partitioning range for datasets. The performance of the forecasting models has been evaluated using *MAPE*.

Considering the results obtained, various conclusion can be deduced. First of all, SVM based models and statistical based Holt-Winters models show better performance than the models developed by other regression methods. The order of the regression methods for 3G/UMTS network traffic forecasting in terms of their prediction performance based on the *MAPE*'s, from the best to the worst, is SVM, Holt-Winters, MLP, RF, and RBF. Secondly, the forecasting models on the daily time scale indicate much better performance than the forecasting models based on the other time scales (that is, hourly and weekly). Thirdly, when the lengths of the time lags are compared, the time lags having longer scales or generated by using autocorrelations yield lower *MAPE*'s on the average while the time lags having shorter scales yield higher *MAPE*'s on the average for hourly traffic forecasting.

Because SVM-based prediction models yield better performance for hourly traffic and Holt-Winters yields the same for weekly traffic, it can be said that SVM is useful for hourly and Holt-Winters for weekly to forecast 3G network voice and data traffic.

Future work can be performed in a number of different areas. Different machine learning methods with different time lags can be applied to forecast the 3G/UMTS network traffic. Additionally, this work can easily be extended to next generation wireless telecommunication technology like fourth generation / Long Term Evaluation (4G/LTE) or fifth generation (5G) mobile networks.

REFERENCES

Akgol, D., Akay, M. F., and Yur, Y., 2015. Performance Comparison of Machine Learning Methods for Network Traffic Forecasting. Third International Symposium on Engineering, Artificial Intelligence & Applications (ISEAIA2015), North Cyprus, 12 – 13.

Ali, J., Rehanullah, K., Ahmad, N., & Maqsood, I., 2012. Random Forests and Decision Trees. International Journal of Computer Science Issues (IJCSI), 9(5), 272-278.

Bai, Y., Ma, K., & Ma, G., 2009. An Analysis of the Combined Wavelet-GM (1, 1) Model for Network Traffic Forecasting. IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC 2009), Beijing, 155-158.

Benzer, R., & Benzer, S., 2015. Application of artificial neural network into the freshwater fish caught in Turkey. International Journal of Fisheries and Aquatic Studies (IJFAS), *2*(5), 341-346.

Broomhead D.S., Lowe D., 1988. Multivariable Functional Interpolation an Adaptive Networks. Complex Systems volume 2, Issue 3, 321 - 355.

Bui, D. T., Pradhan, B., Lofman, O., & Revhaug, I., 2012. Landslide Susceptibility Assessment in Vietnam Using Support Vector Machines, Decision Tree, and Naïve Bayes Models. Mathematical Problems in Engineering, 26.

Chabaa, S., Zeroual, A., and Antari, J., 2010. Identification and Prediction of Internet Traffic Using Neural Networks. Journal of Intelligent Learning Systems and Applications, 2(3), 147 – 155.

Chatfield C., Yar M., 1988. Holt-Winters Forecasting: Some Practşcal Issues. Journal Of the Royal Statistical Society. Series D, Vol.37, No:2, 129-140.

Chen, M., 2011. Short-term forecasting model of web traffic based on genetic algorithm and neural network. In Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 623 – 626.

Chen, X. T., Zhang, S. Y., & Tian, T. T., 2010. Internet Traffic Forecasting Based on BP Neural Network. Journal of Nanjing University of Posts and Telecommunications (Natural Science), 2 – 4.

Chen, Y., Yang, B., & Meng, Q., 2012. Small-time scale network traffic prediction based on flexible neural tree. Applied Soft Computing, 274–279.

Cortez, P., Rio, M., Rocha, M., & Sousa, P., 2006. Internet Traffic Forecasting using Neural Networks. International Joint Conference Neural Networks (IJCNN), 2635 – 2642.

Cortez, P., Rio, M., Souso, P., & Rocha, M., 2007. Topology Aware Internet Traffic Forecasting Using Neural Networks. In Artificial Neural Networks, Porto, Portugal: Springer Berlin Heidelberg. 445 – 454.

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.

Ertekin, Ş., 2009. Learning in Extreme Conditions: Online and Active Learning with Massive, Imbalanced and Noisy Data, Ph.D. Thesis, Pennsylvania, 11-23.

Gowrishankar S., 2009. A Time Series Modeling and Prediction of Wireless Network Traffic. International Journal of Interactive Mobile Technologies (iJIM), 53–62.

Hsiao-Hwa Chen, 2007. The Next Generation CDMA Technologies, John Wiley and Sons, ISBN 978-0-470-02294-8, 105–106.

Kamińska-Chuchmała, A., 2014. Spatial internet traffic load forecasting with using estimation method. Procedia Computer Science, 35, 290-298.

Katris, C., & Daskalaki, S., 2015a. Comparing forecasting approaches for Internet traffic. Expert Systems with Applications, 42(21), 8172–8183.

Katris, C., & Daskalaki, S., 2015b. Combining Time Series Forecasting Methods for Internet Traffic. In Stochastic Models, Statistics and Their Applications, Springer International Publishing, 309-317.

Kim, S., 2011. Forecasting internet traffic by using seasonal GARCH models. Journal of Communications and Networks, *13*(6), 621 - 624.

Kotsiantis, S., Kanellopoulos, D., & Pintelas, P., 2006. Local Boosting of Decision Stumps for Regression and Classification Problems. Journal of Computers, *1*(4), 30-37.

Liu, X., Fang, X., Qin, Z., Ye, C., & Xie, M., 2011. A Short-Term Forecasting Algorithm for Network Traffic Based on Chaos Theory and SVM. Journal of Network and Systems Management, *19*(4), 427-447.

Maurya, C. K., & Minz, S., 2012. Fuzzy inference system for Internet traffic load forecasting. 2012 National Conference on Computing and Communication Systems (NCCCS), 1-4.

Miguel, M. L., Penna, M. C., Nievola, J. C., & Pellenz, M. E., 2012. New models for long-term Internet traffic forecasting using artificial neural networks and flow based information. In Network Operations and Management Symposium (NOMS), 1082 - 1088.

O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee, 2002. Choosing multiple parameters for support vector machines. Machine Learning, 46(1):131-159.

Oliveira, T. P., Barbar, J. S., & Soares, A. S., 2014. Multilayer Perceptron and Stacked Autoencoder for Internet Traffic Prediction. In Network and Parallel Computing, Springer Berlin Heidelberg, 61-71.

Papagiannaki, K., Taft, N., Zhang, Z., & Diot, C., 2005. Long-Term Forecasting of Internet Backbone Traffic: Observations and Initial Models. Neural Networks, *16*(5), 1110 - 1124.

Ratrout, N. T., & Gazder, U., 2014. Factors Affecting Performance of Parametric and Non-parametric Models for Daily Traffic Forecasting. Procedia Computer Science, 32, 285–292.

Taghanaki, S. A., Ansari, M. R., Dehkordi, B. Z., & Mousavi, S. A., 2012. Nonlinear Feature Transformation and Genetic Feature Selection: Improving System Security and Decreasing Computational Cost. Electronics and Telecommunications Research Institute (ETRI) Journal, 34(6), 847-857.

Tan, I. K., Hoong, P. K., and Keong, C. Y., 2010. Towards Forecasting Low Network Traffic for Software Patch Downloads-An ARMA Model Forecast Using CRONOS. Second International Conference on Computer and Network Technology (ICCNT), Bangkok, 88 – 92.

Venkatessh, A., Soundaryadevi, M., & Jayashree, L. S., 2015. Forecasting Energy Demands based on Ensemble of Classifiers. International Journal of Applied Engineering Research (IJAER), 10(5), 13721-13733.

Wang, C., Zhang, X., Yan, H., & Zheng, L., 2008. An Internet Traffic Forecasting Model Adopting Radical Based on Function Neural Network Optimized by Genetic Algorithm. In Conference/Workshop on Knowledge Discovery and Data Mining (WKDD), 367-370.

Wang, Z., & LI, W., 2012. Research on the Network Traffic Time Series Modeling and Forecasting Based on Wavelet Decomposition. Journal of Convergence Information Technology (JCIT), 7(11), 124-131.

Yur, Y., Akay, F., Abut, F., 2016. Short Term Voice Traffic Forecast in 3G/UMTS Networks Using Machine Learning and Statistical Methods, Academics World 41st International Conference, Barcelona, 19-22.

Zhang, Y., & Liu, Y., 2009. Traffic forecasting using least squares support vector machines. Transportmetrica, *5*(3), 93-213.

Zhao, Y., & Zhang, Y., 2008. Comparison of decision tree methods for finding active objects. Advances in Space Research, 41(12), 1955–1959.

3GPP, 1999. Overview of 3GPP Release 99 Summary of all Release 99 Features GSM/EDGE, Phase 2+ Release 99 - See Versions of 3GPP TS 01.01.

3GPP, 2014. Overview of 3GPP Release 11 V0.2.0 (2014-09)

CURRICULUM VITAE

Yasin Yur was born in Aydın-Turkey, in 1975. After completing elementary and high school at Çanakkale, he started a bachelor of science education at the department of Electrical and Electronics Engineering, University of Gaziantep and graduated at 1999. Since then he has been working on telecommunication projects for the global communication companies Siemens, Alcatel, Turk Telekom and PI Works, respectively. In the meantime, he also took part in various abroad projects located in Oman, Bangladesh, Indonesia, and K.S.A. He started MSc program at the department of Computer Engineering since 2015 with Assoc. Prof. Dr. Fatih Akay. He is married and he has a baby boy.