
Advanced Web Traffic Modelling and Forecasting with a Hybrid Predictive Approach

Ujjwal Thakur¹, Sunil K. Singh¹, Sudhakar Kumar^{1,*},
Harmanjot Singh¹, Varsha Arya^{2,3}, Brij B. Gupta^{4,5,6,7,*},
Razaz Waheeb Attar⁸, Ahmed Alhomoud⁹
and Kwok Tai Chui²

¹*Department of CSE, Chandigarh College of Engineering & Technology, Chandigarh, India*

²*Hong Kong Metropolitan University (HKMU), Hong Kong, SAR, China*

³*Center for Interdisciplinary Research, University of Petroleum and Energy Studies (UPES), Dehradun, India*

⁴*Department of Computer Science and Information Engineering, Asia University, Taichung 413, Taiwan, China*

⁵*Department of Medical Research, China Medical University Hospital, China Medical University, Taichung, Taiwan*

⁶*Symbiosis Centre for Information Technology (SCIT), Symbiosis International University, Pune, India*

⁷*University of Economics and Human Science, Warsaw, Poland*

⁸*Management Department, College of Business Administration, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia*

⁹*Department of Computer Science, College of Science, Northern Border University, Arar 91431, Saudi Arabia*

E-mail: ujjwalthakur.008reena@gmail.com; sksingh@ccet.ac.in; sudhakar@ccet.ac.in; srharmanjot2003@gmail.com; varya@hkmu.edu.hk; gupta.brij@ieee.org; Raattar@pnu.edu.sa; aalhomoud@nbu.edu.sa; jktchui@hkmu.edu.hk

** Corresponding Authors*

Received 02 November 2024; Accepted 25 February 2025

Journal of Web Engineering, Vol. 24_3, 409–456.

doi: 10.13052/jwe1540-9589.2434

© 2025 River Publishers

Abstract

Web traffic analysis is crucial for optimising user experience and engagement. This research explores a hybrid approach combining traditional statistical methods, like the autoregressive integrated moving average (ARIMA) model, with advanced techniques such as long short-term memory (LSTM) neural networks and the Prophet model. ARIMA effectively captures linear trends, seasonal effects, and cyclic behaviours, while LSTM handles complex non-linear patterns, and Prophet addresses seasonal variations and missing data. The hybrid model demonstrated 93% accuracy in predicting web traffic, highlighting the benefits of integrating these methodologies. This approach enables businesses to better manage resources, boost user engagement, and improve revenue. Future research will focus on refining hybrid models by incorporating new data features and ensemble methods to further enhance prediction accuracy, ultimately advancing the understanding of web traffic trends and user behaviour.

Keywords: Web traffic analysis, ARIMA, LSTM, prophet model, time series forecasting, predictive analytics, user engagement, seasonal variations, machine learning.

1 Introduction

Web traffic analysis (Ihm and Pai, 2011; Chen and Cheng, 2016) has become increasingly important with the growing popularity and implementation of digital platforms. It includes the regular tracking of users to understand their behaviour on these platforms. There are also other important metrics, such as page views, bounce rates, referral sources, and new versus returning visitors. The collection of this data helps organisations assess the effectiveness of their online campaigns, user engagement, and content navigation. Data collected through web traffic analysis is therefore used for improving user experience (Khade et al., 2012), driving sales, and keeping pace with competitors. In e-commerce, for example, traffic analysis influences product availability or promotion. In digital marketing, web traffic is an important aspect that influences overall conversion rates. Web traffic analysis is also crucial for infrastructure and server agitators. A website needs to know the expected peak user load so that they can invest in timely infrastructure at the hosting week. However, the ability to fathom the ever-increasing influx of web data has made way for newer, more sophisticated models that go beyond

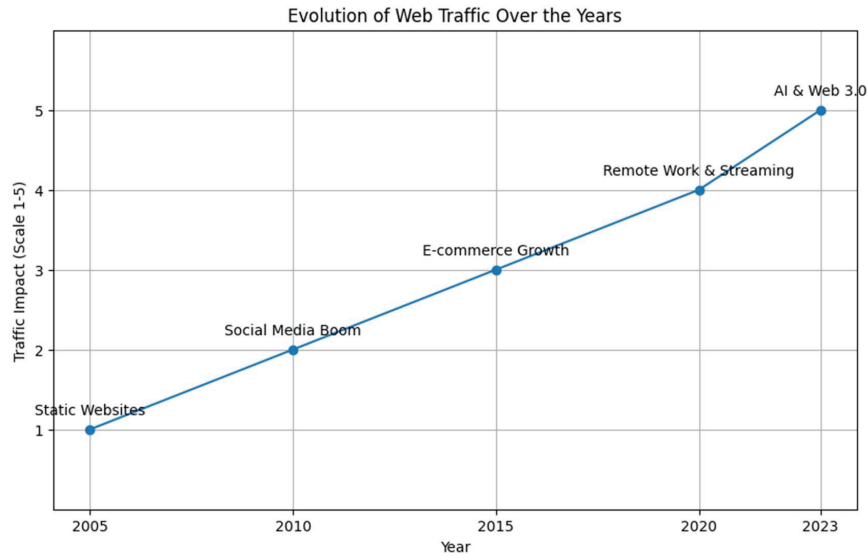


Figure 1 Evolution of web traffic.

basic web traffic analysis. These models can be used to forecast web traffic. One common goal is to go beyond descriptive predictions or just keeping a count of the metrics, and help businesses plan future strategies by giving the forecasts of web traffic. Accurate forecasts can help future scenarios for resource management and can influence real-time context on the website.

The evolution of web traffic over the years is shown in Figure 1. Understanding web traffic forecasting will allow understanding end-user behaviour, provide infrastructure optimisation, and monitor predicted future trends. Businesses deem these predictions necessary for the proper management of events such as Black Friday in order to ensure seamless user experience and avoid traffic jams. By forecasting behaviour in terms of visitor traffic, resource allocation can be adjusted to demand, as traffic will always vary depending on the hours of the day, weekdays, and seasons, allowing operational scaling to work proactively. Recognizing common traffic patterns enables companies to plan their resource allocation; however, traffic may also rapidly increase predictably during events like DDoS attacks or due to viral content. The speed at which these incidents are detected is critical to mitigating the damage caused by them and achieving stabilisation of websites, limiting losses due to downtimes. Viewed as a time series problem, the web traffic forecasting solution faces the challenges of noisy data or gradual shifts

in addition to sudden events that typical models cannot handle effectively. Due to ever-expanding volumes of data, it is becoming increasingly difficult for traditional models to determine the proper patterns, while machine learning is proving to be the perfect solution to a large amount of data developed with much nonlinearity and high dimensionality. Machine learning algorithms learn across huge datasets, which permits them to learn patterns from larger features. Machine learning provides greater automation for web traffic analysis for more accurate predictions. It is proactive in anomaly detection, precisely anticipates high and low spikes in the amount of traffic on websites and makes better decisions based on such findings. This machine learning empowered contemporary way of forecasting also represents a departure from conventional methods. Given the escalating complexity and volume of web traffic, machine learning (ML) models will be a scalable method to manage and classify a variety of factors that influence web traffic patterns ranging from social media forums to global events. By leveraging the potential of ML, organisations can increase attempted forecasting scenarios and adapt work styles to their requirements while also improving builders productivity and operational work efficiency.

1.1 Problems in Web Traffic Prediction

Web traffic is dynamic in nature, affected by various factors such as time, day, month, business promotions, and global events. As a result, predicting web traffic demand for the future becomes a daunting task. For example, one significant problem is seasonality. It is common for a website to be well-tuned to a weekly or monthly cycle in which web traffic has a typical increase and decrease pattern at weekends and clear seasonal trend during holiday seasons. Another important problem is anomalies. Certain unexpected events always occur. For example, a blog post may go viral on social media, or a website may go down during a physical event, or a major disease may break around the world such as the COVID-19 (Gupta et al., 2023) pandemic that immediately affected global web traffic. This type of sudden change is difficult to capture by traditional time series models. In addition, user behaviour is always evolving. As such, some shifts must be observed due to evolving user behaviour, changing technologies, and new competitors. Despite continuous improvement in forecasting methodologies, the difficulty lies in the combination of deterministic and stochastic factors in web traffic demand. Even though web traffic demand increases every weekend and every holiday season, short observed events at non-standard intervals may disrupt

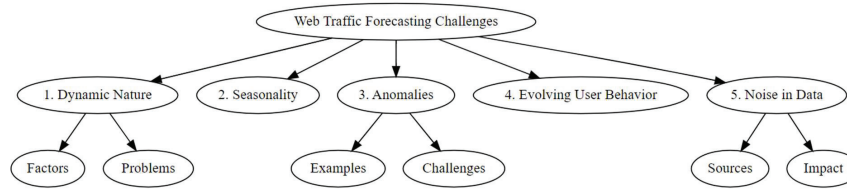


Figure 2 Various problems in web traffic.

the underlying base of the relationship. Furthermore, web traffic datasets are usually overwhelmed by noise. Random spikes often occur, which are mainly caused by accidents, bot promotions, and invalid data, ultimately leading to the distortion of web traffic models. The various problems in web traffic are shown in Figure 2.

Traditional time-series forecasting is limited in modelling nonlinearities and volatility changes, thus acting as a bottleneck in long-term predictions. Although these neural network models, i.e., long and short-term memory (LSTM), are more effective in aligning trends and volatility, they are costly in terms of computation. The huge pace and nature of web data, particularly with frequent updates, make it harder to forecast. Coupled with external influences such as marketing campaigns and news events, this requires adaptive modelling for accurate forecasting. Although mathematical models still need long-term forecasting for computational real-time forecasts, machine learning is a more adaptive and responsive solution.

In this paper, we examine the possibilities of advanced time series modelling methodologies to solve the problem of web traffic forecasting. We focus on three methods: ARIMA (autoregressive integrated moving average) (Yaacob et al., 2010), exponential smoothing, and machine learning (ML) models, especially LSTM (long short-term memory) neural networks, which have shown to provide predictions of time series data with complex structures. Ultimately, we hope to open up opportunities to better understand the strengths and weaknesses of each of these approaches through our comparison of traditional statistical methods, like ARIMA and exponential smoothing, alongside machine-learning approaches in this realm. ARIMA is an established time series forecasting tool because of its ability to model short-term dependence in the data and generate estimates based on trends and seasonality. However, ARIMA does have limitations when modelling non-linear traffic and/or non-regular traffic patterns. Exponential smoothing, specifically Holt-Winters exponential smoothing, provides another lens to view data analysis, focusing on more recent data points and adjusting for

seasonality, making it a more reactive perspective on time series data than ARIMA.

Conversely, machine learning approaches (Mengi et al., 2023; I. Singh et al., 2022), such as LSTM networks, have entered the realm of time series data because of their capacity to account for complex, non-linear patterns found in real world time series data. The LSTM architecture is able to “learn” and “remember” certain patterns for from days to years and is less impacted (or noisy) by moving averages and sudden spikes in traffic. While computationally more expensive, time-series machine learning models provide an exciting area of growth for web traffic forecasting, especially when paired with large datasets and feature engineering.

This paper will also include a comparison in the evaluation of these approaches through real world web traffic data that gauges their performance and all three methods with overall metrics, such as MAE (mean absolute error); RMSE (root mean squared error); and MAPE (mean absolute percentage error).

The goal is to provide a comprehensive analysis of the accuracy, computational efficiency, and scalability of these methods when applied to web traffic forecasting.

The remainder of this paper is organised as follows:

Section 2, Literature Review, investigates relevant studies/statistics literature on web traffic analysis and forecasting techniques. It gives an overview of the ARIMA, exponential smoothing methods, and recent developments in machine learning models, especially in forecasting time series. In Section 3, Mathematical Models and Methodology, we will discuss the mathematical foundations of ARIMA and exponential smoothing models as well as the architecture of LSTM networks. In this section, we present core equations for each model, discuss model parameter selection and model evaluation error metrics. In Section 4, Experimental Setup, we present the experimental design that we used to evaluate models. In this section, we provide a detailed description of the dataset, and explain the applied preprocessing of the data and the training/test processes we used for the modelling. This section will also detail the software we implemented for the libraries for each modelling. In Section 5, Results and Analysis, we discuss each model’s performance. In this section, we will provide two line graphs showing the actual and the predicted traffic patterns, and a table of checked model accuracy per evaluation metric and limitations are discussed in Section 6. Section 7 discusses applications of the results from the analysis, the strengths and weaknesses of each model, and possible uses of the findings in practice. In Section 8, we

address some of the possibilities for future research, including hybrid models that combine statistical methods with machine learning. We also discuss potential benefits of further research in forecasting. Lastly, in the concluding section, an overview of the main results of the study and a reflection on the contribution of modern methods for web traffic analysis and forecasting will be presented. This includes applications of these findings in practice and suggestions for future research.

2 Literature Review

2.1 Conventional Frameworks for Web Traffic Forecasting

Web traffic forecasting has for a long time been a central task in the management of online services, and many traditional models have been developed over the years. Traditional models are typically characterised by a statistical time series framework that attempts to capture the temporal structure in traffic data. The traditional time series methods are usually based on popular methods such as ARIMA (autoregressive integrated moving average) (El Hag and Sharif, 2007), Holt-Winters exponential smoothing, and SARIMA (seasonal ARIMA). Table 1 shows each model's key features and limitations.

1. **ARIMA:** ARIMA models, along with Holt-Winters, are some of the most widely-used models in time series forecasting. ARIMA models account for trends by combining autoregression and moving average with integration to manage the non-stationarity of the data. While ARIMA models are reliable approaches to forecast linear data and have success in short-term forecasting, they are often found ineffective when it comes to long-term forecasting, seasonality and nonlinear trends in data. The ARIMA model architecture is shown in Figure 3.
2. **Holt-Winters exponential smoothing:** This model is appropriate for data that has both trend and seasonality components. Holt-Winters updates simple exponential smoothing by adding trend and seasonal components, but, like ARIMA, Holt-Winters typically assumes a linear pattern in data and struggles when data has a sudden change or nonlinear changes. The architecture of Holt-Winters is shown in Figure 4.
3. **SARIMA:** The SARIMA (Samal et al., 2019; Yuan et al., 2013) model extends ARIMA by accounting for the possibility of seasonal differencing in addition to non-seasonal differencing. SARIMA is useful for periodic traffic patterns but like ARIMA does not account adequately for complex, nonlinear or chaotic traffic behaviours.

Table 1 Key features and limitations of model used

Model	Key Features	Limitations
ARIMA	Captures linear relationships, uses autoregression and moving averages	Struggles with seasonality, nonlinearity
Holt-Winters	Accounts for trend and seasonality, exponential smoothing	Limited to linear trends, not adaptive to sudden changes
SARIMA	Seasonal component included, useful for periodic patterns	Poor performance with irregular, chaotic data

ARIMA Model Architecture Overview

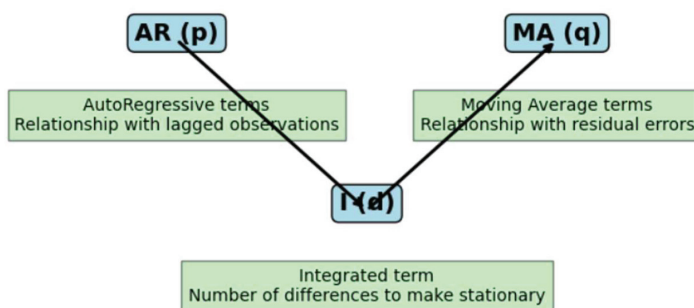


Figure 3 Overview of ARIMA.

Holt-Winters Exponential Smoothing Architecture Overview

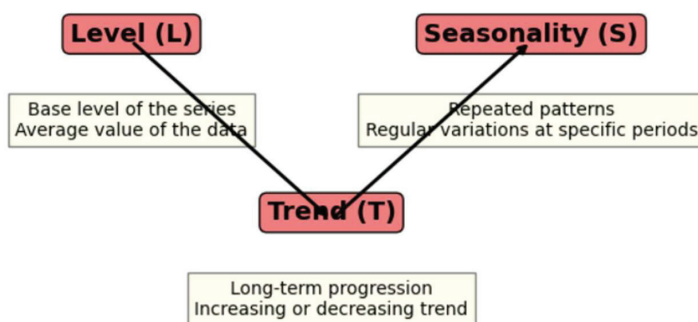


Figure 4 Holt-Winters overview.

Despite being the cornerstone of time series forecasting, the assumptions of data linearity and stationarity within which these models operate hampers their usability on contemporary web traffic data, which are usually volatile and nonlinear. In addition, these models require tedium, often requiring a fair amount of manual tuning, and are not particularly efficient or effective with larger datasets.

2.2 Machine Learning-based Approaches

1. Following the emergence of machine learning (ML) (Kaur et al., n.d.; Sharma et al., 2023; S. Gupta et al., 2023), industry practitioners forecast web traffic using models that more readily identify nonlinear patterns, more complex interactions, and hidden relationships in the data. ML (Shewalkar et al., 2019) provides practitioners with some advantages from traditional time-series forecasts due to these capabilities. First, ML has a better proficiency with larger datasets. Secondly, ML provides practitioners better flexibility, as web traffic can change constantly over time. Thirdly, ML models are capable of incorporating some exogenous features into forecasting if desired.
2. Random forests and gradient boosting machines (GBMs): Some promising results in time series forecasting are being reported using ensemble learning methods like random forests and GBMs. In an ensemble learning model predictions are made based on combining a number of weak learners or trees together, sequentially over time, producing more accurate forecasts, relatively speaking. While random forests and GBMs are highly effective at capturing relationships that are complicated and nonlinear, the models are not designed specifically for sequential or time-dependent data and many require additional feature engineering, e.g., incorporating lag-based features, into their engineering structure in order to provide good judgement in a forecasting event.
3. Support vector machine (SVM): Support vector machine regression (SVR) (B.-J. Chen et al., 2004) is another popular framework where the SVM model structure can also be used for time-series forecasting. The SVM model structure has been shown to efficiently construct a forecast as it uses a margin that will only provide a fit between support vectors maximising the margin. Thus, SVM performs well in the presence of noise in the outcome variable. However, SVMs can only handle data that are not too large, and will not perform as well with large-scale web traffic data replicating the traditional time series methodology due to computational complexity.

4. Deep learning models: Long short-term memory (LSTM): LSTMs, which are recurrent neural networks (RNNs) (Hasnain et al., 2019; Belavadi et al., 2020) are considered to be a paradigm shift in the field of time series forecasting. The LSTM network serves as an effective approach to estimate long-term dependencies, which are more effective for datasets with complicated nonlinear web traffic patterns. The LSTM's selective ability to remember or to forget past information attributes to its overall effectiveness due to the variety of different patterns, both short-term and long-term, related to forecasting web traffic. However, an LSTM requires copious amounts of data to be trained effectively and is computationally expensive.
5. Prophet: Prophet (Katwal et al., 2024; Belavadi et al., 2020) is a newly developed time series forecasting model by Facebook to provide a natural approach to estimating seasonality as well as holidays. This is particularly useful in forecasting web traffic datasets because it will add additional capabilities to detect outliers within a dataset, and fit nonlinear trends. Prophet serves as a simple yet effective methodology to use with business-related time series datasets as an alternative to more complicated ML forecasting models. However, Prophet models are not as effective as deep learning methods when it comes to highly complex nonlinear datasets. The architecture of the Prophet model is shown in Figure 5.

Table 2 shows the different ML models which can be used for web forecasting. They have features and limitations.

2.3 Comparative Performance of Traditional vs. Machine Learning Approaches

Studies have shown that while traditional models such as ARIMA and SARIMA perform reasonably well for short-term forecasting and stationary data, they fall short in terms of accuracy and scalability when applied to more dynamic and complex web traffic datasets. Machine learning models, especially deep learning approaches like LSTM, outperform traditional models in capturing long-term dependencies and handling nonlinearity. However, machine learning models often require more computational resources and are more difficult to interpret, which can be a challenge in practical applications.

Table 3 highlights the relative performance of traditional and machine learning models based on common metrics used in forecasting, such as root mean squared error (RMSE) and mean absolute percentage error (MAPE).

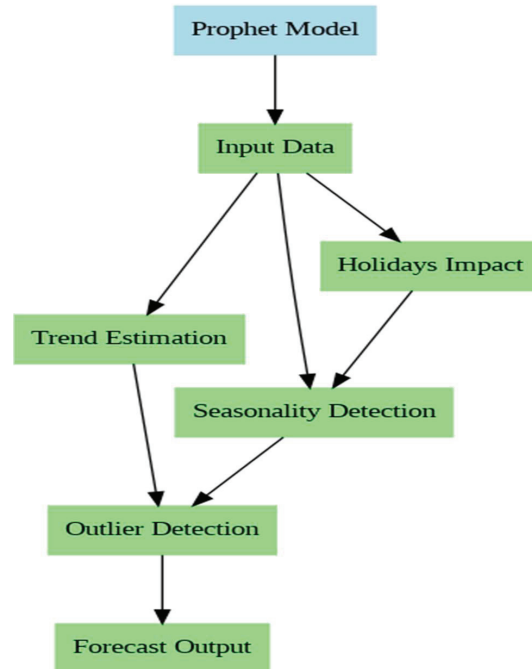


Figure 5 The Prophet model.

Table 2 Features and limitations of ML model used for web forecasting

ML Model	Key Features	Limitations
Random forests, GBMs	Effective for nonlinear patterns, ensemble learning	Not inherently time-series focused, requires feature engineering
SVM (SVR)	Robust against noise, margin-based regression	Computationally expensive for large datasets
LSTM	Captures long-term dependencies, excellent for time-series forecasting	Requires large datasets, high computational cost
Prophet	Handles seasonality, holiday effects, easy to use	Less effective for highly complex patterns compared to deep learning models

As seen in Table 3, LSTM models tend to deliver the best accuracy (lowest RMSE and MAPE), but their complexity and need for high computational resources make them harder to implement in some cases. ARIMA, while easy to interpret, has high error rates compared to ML models, especially in non-linear traffic patterns.

Table 3 Advantages and disadvantages of each model

Model	RMSE (Lower is Better)	MAPE (Lower is Better)	Scalability	Complexity	Interpretability
ARIMA	High	High	Low	Low	High
SARIMA	Medium	Medium	Low	Medium	High
Random forests	Medium	Medium	Medium	High	Medium
SVM	Low	Low	Low	Medium	Low
LSTM	Lowest	Lowest	High	Very High	Low
Prophet	Medium	Medium	High	Medium	High

Table 4 Hybrid models key features and limitations

Hybrid Model	Key Features	Limitations
ARIMA-LSTM	Combines ARIMA for linear trends with LSTM for nonlinear patterns	High complexity, computationally expensive
SARIMA-Prophet	Uses SARIMA for seasonal trends and Prophet for holidays/outliers	May not generalise well to highly volatile traffic patterns

2.4 Hybrid Models: Combining Traditional and Machine Learning Approaches

Recent studies have also explored hybrid models that combine traditional statistical methods with machine learning techniques to leverage the strengths of both. For example, hybrid ARIMA-LSTM (Yamak et al., 2020) models use ARIMA to model linear trends and LSTM to handle nonlinear residuals, resulting in improved forecasting performance. These models are designed to address the shortcomings of both approaches, offering a balance between accuracy, complexity, and interpretability, as shown in Table 4.

While hybrid models show promise in improving forecasting accuracy, they introduce new challenges in terms of model complexity, parameter tuning, and implementation. As these models require combining different methodologies, they also demand more computational resources and expertise, making them harder to implement for large-scale, real-time forecasting tasks.

2.5 Gaps in Existing Research

Despite significant advancements, several gaps remain in the current research landscape of web traffic forecasting:

1. **Scalability and real-time forecasting:** Many existing models struggle with scalability when applied to large datasets or real-time forecasting

environments. The computational cost of training complex models like LSTMs or hybrid models can make real-time forecasting infeasible for large-scale web services.

Proposed solution: The hybrid model uses ARIMA for efficient linear trend processing and Prophet for fast seasonal adjustments, reducing the computational burden. This model assigns differentially complex nonlinear features to LSTM and thus shall strike a balance between computation time and scalability, fitting better into the real-time paradigm.

2. **Handling abrupt changes:** While machine learning models like LSTM can handle nonlinear patterns, they often fail to account for abrupt changes or anomalies in traffic caused by external events (e.g., viral trends, marketing campaigns). Incorporating external factors into models remains an underexplored area in research.

Proposed solution: LSTM works well at modelling nonlinear dependencies and long-term trends. Prophet adjusts for special events and anomalies like holidays or traffic spikes in order to be robust to sudden changes.

3. **Interpretability:** Machine learning models, especially deep learning ones, are often considered “black boxes” because of their lack of transparency. While traditional models are more interpretable, the high complexity of ML models makes it difficult for users to understand the reasons behind specific forecasts, which can be a barrier in decision-making.

Proposed solution: The interpretability from ARIMA and Prophet was achieved by using linear trend and seasonality, while LSTM helped increase prediction accuracy.

These promises ensure that they deliver stakeholder value with insight and forecasting accuracy.

4. **Hybrid model optimisation:** Although hybrid models combine the strengths of traditional and ML approaches, optimising these models remains challenging. More research is needed to streamline the integration of statistical and machine learning techniques to build models that are both accurate and computationally efficient.

Proposed solution: The proposed model integrates the simplicity of ARIMA and Prophet into the complexity-handling capabilities of LSTM. Model parameter optimisation is conducted with validation measures like RMSE and MAPE to enable streamlined integration.

5. **Non-linearity and discontinuity:** Traditional models such as Holt-Winters or SARIMA fail to capture the nonlinear patterns and instant

changes due to external events (like viral content or promotions). Due to this reason, the combination of the hybrid model uses LSTM to capture complex nonlinear dependencies while Prophet detects and accounts for outliers and seasonal interruptions. Hence, they improve robustness towards the change and anomalies.

Proposed solution: LSTMs can model disjoint patterns with abrupt changes while Prophet can model standard seasonal changes and events. This complementary combination ensures robustness in response to unpredictable behaviours.

6. **Manual parameter tuning with conventional models:** Traditional models such as ARMA and SARIMA rely on extensive manual parameter tuning which can be labour-intensive and prone to human errors. Machine learning approaches such as random forest and gradient boosting machines rely heavily on feature engineering, such as manually creating lag features or encoding seasonality. Most studies have focused on specific datasets and have not confirmed that their methods can be generalised to other domains or industries.

Proposed solution: LSTMs do not require many engineering measures because they automate learning processes from data. Prophet simplifies seasonal adjustments while minimizing manual adjustments.

7. **Real-time applicability:** Current models are very limited in the real-time predictions they can make due to a very high expense in computation time or through extensive manual parameter tuning. Due to this reason, the hybrid model employs ARIMA in lightweight pre-processing and Prophet for fast seasonal adjustment, hence reducing the computational burdens, making the model more amenable to real-time prediction. Proposed solution: The hybrid model thus leverages ARIMA and Prophet for light pre-processing and speedy adjustments making it viable for real-time deployments.

2.6 Mathematical Modelling of Web Traffic

Web traffic analysis relies on understanding the underlying characteristics of traffic patterns and applying mathematical models that can predict future behaviour. These models must account for time series properties like seasonality, trends, noise, and stationarity while making use of various techniques, such as statistical models and machine learning approaches. In this section, we will delve into the critical components of mathematical modelling of web traffic, including time series characteristics, statistical models, hybrid models, and data preprocessing.

2.6.1 Time series characteristics of web traffic

Web traffic data (Syu et al., 2017) typically exhibit several key properties that make forecasting challenging. Recognizing these properties is essential for developing accurate models. The four primary characteristics are:

1. **Seasonality.** Seasonality refers to repeating patterns or cycles observed in the data at regular intervals. Web traffic frequently exhibits daily, weekly, or monthly seasonal patterns due to user behaviour, such as peak activity during business hours or certain days of the week. Seasonal components can be captured mathematically by incorporating sinusoidal functions or seasonal differencing in time series models.
 - **Daily seasonality:** For many websites, traffic peaks during specific hours of the day and is lower during the night. This cyclic behaviour is commonly observed in e-commerce, social media, and educational websites.
 - **Weekly seasonality:** Some sites, such as business platforms, experience higher traffic on weekdays compared to weekends.
 - **Monthly/annual seasonality:** Certain sectors, such as retail, exhibit seasonal patterns linked to holidays or promotional events (e.g., Black Friday, Christmas).
2. **Trend.** The trend represents the long-term movement of the data, which could be upward or downward over time. In web traffic analysis, a positive trend might indicate growing website popularity, while a negative trend could signal a decline. Understanding these trends is vital to accurately modelling and forecasting long-term traffic growth or decay.
 - **Linear trend:** An increase in traffic over time, represented by a linear slope.
 - **Exponential trend:** When traffic grows at an increasing rate, often requiring transformation before applying linear models.
3. **Noise.** Noise in web traffic data refers to random fluctuations or irregular patterns that cannot be explained by trends or seasonality. Noise can be caused by unpredictable events such as server downtimes, sudden surges due to viral content, or irregular user activity.
 - **Random walk:** A process where future values depend on current values plus a random error term. This often represents the unpredictable, stochastic part of the traffic data.
4. **Stationarity.** Stationarity is a key assumption in many time series models, referring to the statistical properties of the series (such as

mean and variance) remaining constant over time. Web traffic data are often non-stationary due to the presence of trends and seasonality. Therefore, stationarity must often be achieved through techniques such as differencing, detrending, or transforming the data.

- **Stationary time series:** After removing trends and seasonality, a time series becomes stationary if its statistical properties do not change over time.
- **Non-stationary time series:** Exhibits varying statistical properties, making direct forecasting challenging without transformation.

3 Mathematical Models for Web Traffic Forecasting

3.1 Statistical Models

3.1.1 ARIMA (autoregressive integrated moving average)

ARIMA is one of the most widely used models in time series forecasting. It combines autoregression (AR), moving averages (MA), and integration (I) to model the temporal dependencies in data. ARIMA models work well for data without a seasonal component and are especially suited for univariate time series data.

- **Autoregression (AR):** Predicts the current value of the series based on past values.
- **Integration (I):** Differencing the data to make it stationary.
- **Moving average (MA):** Models the error of the predictions based on previous error terms.

The general form of ARIMA(p, d, q) includes:

- p: The number of autoregressive terms.
- d: The degree of differencing needed to make the series stationary.
- q: The number of moving average terms.

Equation for ARIMA:

$$Y_t = c + \phi_1 Y_{t-1} + \cdots + \phi_p Y_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \quad (1)$$

where:

- Y_t : Predicted value at time t
- C: Constant
- Φ : AR coefficients
- Θ : MA coefficients
- ε_t : Error terms.

3.1.2 SARIMA (seasonal ARIMA)

SARIMA extends the ARIMA model to account for seasonality by adding seasonal autoregressive, seasonal differencing, and seasonal moving average components. This makes it well-suited for web traffic data that show repeating seasonal patterns. The general form of SARIMA (p, d, q) (P, D, Q, s), where P, D, and Q represent seasonal components and s is the seasonal length, models traffic with cycles such as daily or weekly patterns. Equation for SARIMA:

$$(1 - B)(1 - \Phi B^s)Y_t = c + (1 + \theta B)(1 + \Theta B^s)\varepsilon_t \quad (2)$$

where:

- B: Lag operator
- Φ : Seasonal autoregressive parameters
- Θ : Seasonal moving average parameters
- s: Seasonal length (e.g., 24 for daily data).

3.1.3 Hybrid models

Hybrid models combine statistical techniques with machine learning algorithms to leverage the strengths of both approaches. They often result in more accurate and robust predictions by combining the interpretability of statistical models with the ability of ML to capture complex, nonlinear patterns.

1. **ARIMA-LSTM.** In an ARIMA-LSTM hybrid model, ARIMA is first used to model the linear components in the data. The residuals (errors) from ARIMA are then fed into an LSTM (long short-term memory) model to capture any remaining nonlinear patterns or dependencies in the data. This approach allows for better accuracy when web traffic data contain both linear trends and nonlinear dependencies.
2. **SARIMA-Prophet.** SARIMA captures the seasonal and linear trends, while Prophet, developed by Facebook, captures holidays and other outliers. This hybrid approach is ideal for web traffic with known periodicity and sporadic spikes, such as during promotions or major events.

3.2 Strategic Model Selection: A Tailored Approach

ARIMA. With a proven performance in forecasting time series data, particularly univariate data where relationships between observations lagged in time present themselves strongly, it is the model of choice to apply forecasting

of time series data. It captures short-term linear dependency by integrating autoregression (the AR), moving averages (MA), and differencing (I). It is best applicable to stationary data that does not change variably within time periods. Due to their computationally light nature, these the models are applicable in cases where quick outputs are a requirement, especially in resource-limited scenarios. The high interpretability nature allows stakeholders to understand the effect of trends and seasonality on forecasted values, thus making it acceptable in business contexts.

Limitations addressed in the hybrid approach: ARIMA alone struggles with nonlinear patterns which cause sudden surges in traffic. The hybrid model will address the shortcomings by incorporating them along with LSTM and Prophet.

LSTM. Long short-term memory (LSTM) networks, a special kind of recurrent neural network (RNN), are designed to learn sequential dependencies within data. LSTMs have proven capabilities in grasping nonlinear trends and anomalies in web traffic, like sudden spikes from viral content or disruptions posed by external events (promotion activities). It can remember dependencies over long stretches of time, which is key to forecasting future web traffic, due to the influencing effects of accumulating historical patterns. Traditionally, it was called robustness to noise and anomaly; web traffic data is often messy with noise and missing values. The ability of LSTM to learn complex patterns makes it robust to cope with them, especially with pre-processing (such as normalisation and imputation) being done. Web traffic often shows highly nonlinear behaviour owing to user interactions, marketing strategies, and external disruptions. Here, LSTM offers the kind of flexibility that will allow its effective modelling of such dynamic, high-dimensional data. Despite being extremely heavy on computation, it, however, learns patterns without requiring extensive manual features engineering.

Why not GRUs or simpler RNNs? Gated recurrent units (GRUs), simpler alternatives to LSTM, can outperform it on certain occasions. GRUs possess a simpler memory cell structure, which may serve to hinder long-term dependency retention. LSTM actually provides a better alternative for datasets with complex seasonality and high variability. Old-fashioned RNNs fall prey to problems like diminishing gradient, which renders them less effective to model long-term dependencies against LSTM.

Limitations addressed in the hybrid approach: Within the hybrid model, the computational cost for LSTM is lowered since the traditionally simplistic

operations of seasonal adjustment and linear trend modelling are delegated to ARIMA and Prophet. This allows LSTM to focus exclusively on learning nonlinear dependencies.

Prophet. Prophet is essentially a modified version of a more featured integration with Facebook's suite, aimed specifically at time series data characterised with inherent trends, seasonality, and holiday effects. Due to its nature, this tool discovers and combines the automatic production of seasonal variations, it works most effectively for calendars, such as web traffic, with repeat visits during certain time intervals on a single day. It can reasonably adjust outliers and sudden changes, making it reasonably robust against anomalies. It requires minimum manual intervention and hence it is quite easy to use as compared to ARIMA or SARIMA. It is computationally lightweight in terms of time trade, making it good for predictions in comparatively quick times yet of reasonable accuracy in shorter-to-midrange forecasts.

Limitations addressed in hybrid approaches: Prophet appears to work inefficiently on nonlinear dependencies or long seasonality, something which is better handled by LSTM.

Using Prophet in conjunction with LSTM will enable the model to reap the best of both worlds while rectifying each other's flaws.

The choice of the hybrid model realizing the integration of ARIMA, LSTM, and Prophet is drawn from their compatibility:

- ARIMA and SARIMA provide powerful modelling for linear trends with stationary seasonal components, contributing to interpretability and computational efficiency.
- LSTM extracts the nonlinear dependence, long-term trends, and complex patterns of web traffic to assure accuracy in dynamic cases.
- Prophet allows adjustments for seasonality, holidays, and anomalies, thereby reinforcing disruptions in the data.

4 Experimental Setup and Implementation

We present the experimental design that we used to evaluate models. In this section, we provide a detailed description of the dataset, explain the applied preprocessing of the data, and the training/test processes we used for the modelling. This section will also detail the software we implemented for the libraries for each modelling. Figure 6 shows a flowchart of the working or flow of the whole model.

4.1 Data Preprocessing

Effective data preprocessing is crucial for any forecasting model to perform well. Raw web traffic data is often noisy, incomplete, and contains anomalies, requiring several steps to clean and prepare the data for analysis.

1. **Feature extraction.** Feature extraction involves selecting and creating new variables from the raw data that are relevant to the forecasting task. For web traffic, important features include:
 - **Lagged features:** Previous time steps in traffic.
 - **Rolling statistics:** Moving averages and moving standard deviations.
 - **Time features:** Day of the week, hour of the day, month, etc.
 - **External factors:** Events, promotions, holidays that affect traffic.

Feature extraction is essential for machine learning models, which often require a set of input features beyond the raw traffic counts.

2. **Normalisation.** Web traffic data can span several orders of magnitude, especially if it includes a mix of low-traffic and high-traffic days. Normalising the data scales it to a consistent range (typically $[0, 1]$ or $[-1, 1]$), which ensures that models treat all features equally and improves convergence during training.

Min-Max normalisation:

$$X' = \frac{(X - X_{\min})}{(X_{\max} - X_{\min})} \quad (3)$$

Z-score normalisation:

$$X' = \frac{(X - \mu)}{\sigma} \quad (4)$$

where μ is the mean and σ is the standard deviation.

4.2 Handling Missing Data

Missing data is a common problem in web traffic analysis, caused by server downtimes, tracking issues, or other anomalies. To handle missing data, several strategies can be applied:

- **Imputation:** Replace missing values using statistical methods such as mean, median, or mode imputation, or more advanced techniques like forward and backward filling.
- **Interpolation:** For time series data, linear interpolation or spline interpolation can be used to estimate missing values.

4.3 Outlier Detection and Removal

Outliers in web traffic data can be caused by rare events such as viral content or system failures. If not handled properly, these outliers can skew the model's predictions.

- **Z-score method:** Detects outliers based on how many standard deviations away a data point is from the mean.
- **IQR (interquartile range):** Defines outliers as points that lie outside 1.5 times the IQR above the third quartile or below the first quartile.

Algorithm: The following algorithm describes how to forecast Wikipedia article traffic using a hybrid model that combines ARIMA, LSTM, and Prophet. Each model is used to exploit different features of the time-series data (e.g., linear trends, non-linear patterns, and seasonality). After training each model separately, the final predictions are combined using a weighted average to produce a more accurate forecast.

Step 1: Preprocessing

- **Input:**
 - Dataset: Time-series data with daily web traffic for Wikipedia articles from July 1, 2015, to December 31, 2016 of a specific topic there are different articles from the same topic.
 - Each row represents a time series for a specific Wikipedia page.

Preprocessing tasks:

1. **Handle missing values:**
 - Use forward-fill (ffill) and backward-fill (bfill) techniques to fill missing data points.
 - Forward-fill: If a data point is missing, replace it with the previous day's traffic.
 - Backward-fill: If a missing value remains, replace it with the next day's traffic.
2. **Data normalisation (for LSTM):**
 - Normalise the time series between 0 and 1 using Min-Max scaling for better performance in LSTM.
 - Formula:

$$X' = \frac{(X - X_{\min})}{(X_{\max} - X_{\min})}$$

where X' is the normalised data, X is the original value, and X_{min} , X_{max} and are the minimum and maximum values of the time series.

Step 2: ARIMA model (autoregressive integrated moving average)

- **Goal:** Capture linear dependencies in the data using ARIMA.
- **Process:**
 1. **Select Parameters:** Define the ARIMA model with parameters p , d , and q .
 - p : Number of autoregressive terms.
 - d : Degree of differencing (to make the series stationary).
 - q : Number of moving-average terms.
 2. **Train the ARIMA model:**
 - Fit the ARIMA model on the training dataset (first N days).
 - Example ARIMA model order: ARIMA(5,1,0), where:
$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_5 y_{t-5} + \epsilon_t \quad (5)$$
 3. **Forecast future values:**
 - Use the trained model to forecast the next 30 days (test period).

Step 3: LSTM model (long short-term memory)

- **Goal:** Capture non-linear relationships and long-term dependencies in the time series using an LSTM neural network.
- **Process:**
 1. **Data preparation:**
 - Reshape the normalised data into sequences where each sequence contains the last 100 days as input to predict the next day.
 - Example:
$$X(t) = \{X(t-60), X(t-59), \dots, X(t-1)\} \quad (6)$$
 - **Build LSTM model:**
 - Define a Sequential model with two LSTM layers followed by a Dense layer for final prediction.

- Example architecture:
 - Layer 1: LSTM with 50 units (returning sequences).
 - Layer 2: LSTM with 50 units.
 - Output layer: Dense layer with 1 unit.
- **Train the LSTM model:** Train the model using the prepared training data (60-day sequences). Use mean squared error (MSE) as the loss function.
- **Generate future predictions:** Use the last 100 days of the training data as input and recursively predict the next 30 days and so on.

Step 4: Prophet model (Facebook Prophet)

- **Goal:** Capture seasonality and holiday effects using Prophet, a model designed for time series with strong seasonal patterns.
- **Process:**
 1. **Prepare data for Prophet:**
 - Create a dataframe with two columns: ds (date) and y (traffic values).
 2. **Fit the Prophet model:**
 - Train the Prophet model on the historical data, specifying daily seasonality.
 3. **Forecast future values:**
 - Create a future dataframe for the next 30 days.
 - Use the Prophet model to forecast traffic values for the next 30 days.

Step 5: Combine predictions

- **Goal:** Combine the predictions from ARIMA, LSTM, and Prophet to create a robust forecast.
- **Process:**
 1. **Assign weights:**
 - Assign weights to the predictions from each model based on their accuracy on validation data. For example:
 - ARIMA: $w_1=0.4$
 - LSTM: $w_2=0.3$

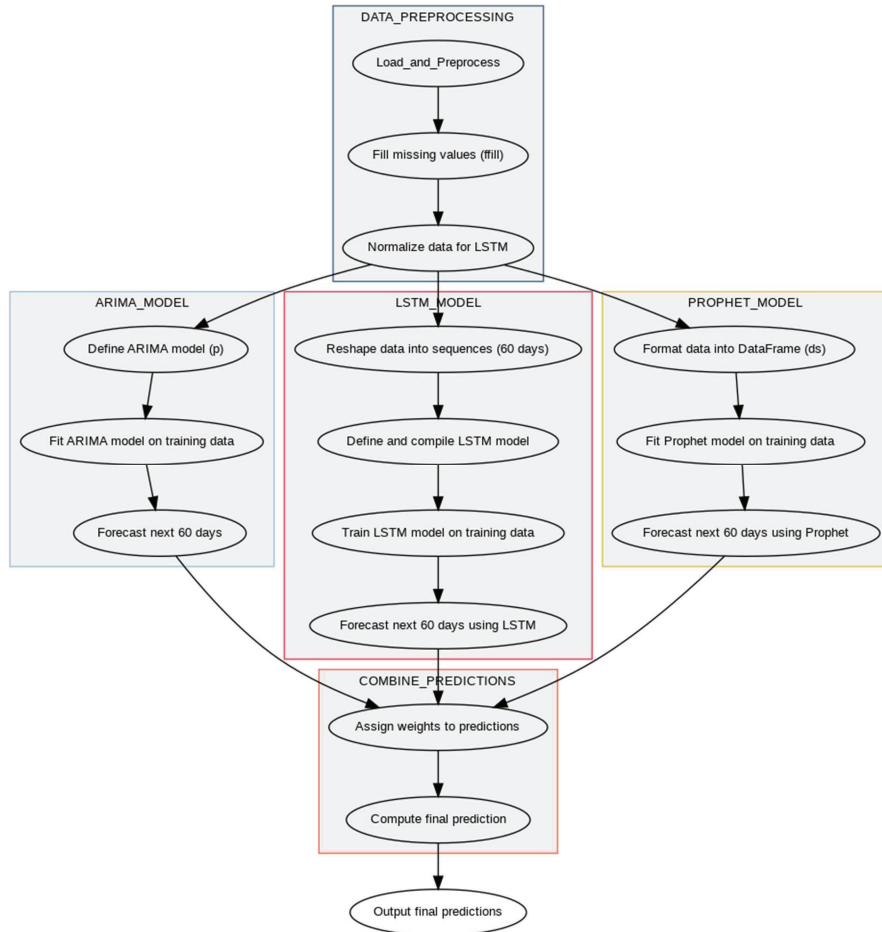


Figure 6 Flow chart of model.

■ Prophet: $w_3=0.3$

2. **Weighted average:** Compute the final prediction as a weighted average of the individual model predictions:

$$\begin{aligned}
 Final_pred(t) &= w_1 \cdot ARIMA_pred(t) \\
 &\quad + w_2 \cdot LSTM_pred(t) \\
 &\quad + w_3 \cdot Prophet_pred(t) \quad (7)
 \end{aligned}$$

- **Output:** The final prediction for each article's traffic over the next 30 days.

4.4 Working of the Hybrid Time Series Forecasting Model

This hybrid model integrates ARIMA, LSTM, and Prophet – three popular time-series forecasting models – to predict Wikipedia article traffic. Each model brings its unique strengths, allowing the hybrid approach to handle linear trends, non-linear patterns, long-term dependencies, and seasonal effects. The result is a robust model that can make accurate forecasts by combining the best aspects of these techniques. In this paper, we will discuss the overall workflow of the model, how each individual component operates, and how their predictions are integrated to generate the final forecast, as depicted Figure 7.

4.4.1 Overview of time series forecasting

Time series forecasting is a common machine learning task where the objective is to predict future values of a given dataset based on historical data points. The key challenge lies in capturing trends, seasonality, and irregularities from past behaviour. This is particularly important for the Wikipedia article traffic dataset, which consists of traffic data over time for various articles, requiring models that can detect both short-term and long-term patterns.

To address these challenges, three models are employed:

- **ARIMA:** Captures linear trends and short-term dependencies.
- **LSTM:** A type of neural network that excels at handling long-term dependencies and non-linear relationships.
- **Prophet:** A tool designed for time-series forecasting with seasonality, trend, and holiday components.

The hybrid model improves forecasting accuracy by combining the strengths of these three models and reducing the weaknesses of any single approach.

4.4.2 Preprocessing the dataset

Before applying the models, the dataset undergoes preprocessing. The dataset contains daily web traffic for various Wikipedia articles from 1 July 2015 to 31 December 2016. Each row corresponds to a specific article, and columns represent daily traffic data. Missing values are prevalent in the dataset, which can either represent zero traffic or unreported data.

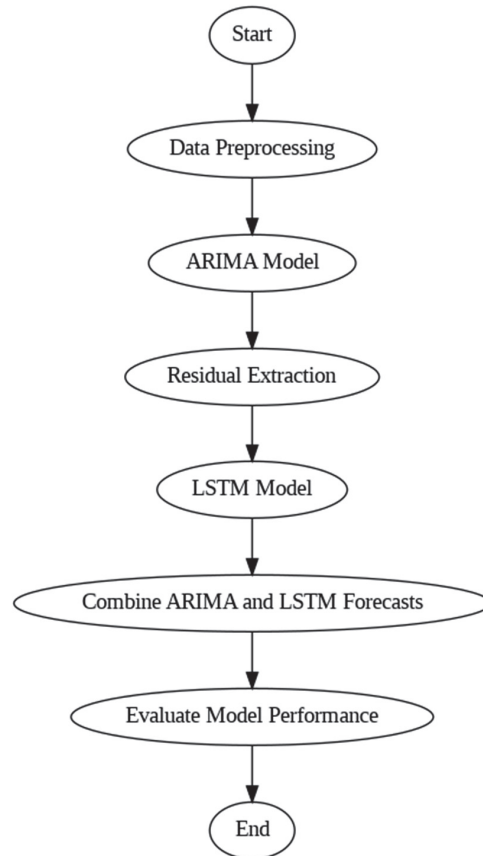


Figure 7 Flowchart of model coming from LSTM and ARIMA.

The preprocessing steps are:

- Filling missing values: Missing values are filled using forward-fill (ffill), which replaces missing values with the previous day's traffic. If a missing value remains after forward-fill, it is replaced using backward-fill (bfill), which uses the next available traffic value.
- Normalisation (for LSTM): For LSTM, the data is normalised using Min-Max scaling. This is necessary because neural networks perform better when input data is normalised, typically scaling values between 0 and 1. Min-Max scaling ensures that the LSTM can effectively learn the patterns in the data without being overwhelmed by the magnitude of values.

4.4.3 Model components

4.4.3.1 *ARIMA model (autoregressive integrated moving average)*

ARIMA is a classical time-series forecasting model that is widely used for linear forecasting. It works by combining three components:

- AR (autoregressive): A component that uses the relationship between an observation and a number of lagged observations (previous time steps).
- I (integrated): Differencing of raw observations to make the time series stationary, which helps remove trends and seasonality.
- MA (moving average): A model that uses dependency between an observation and a residual error from a moving average model applied to lagged observations.

The ARIMA model captures linear dependencies in the time series, which makes it suitable for datasets that exhibit consistent short-term patterns or trends. In this hybrid model, ARIMA is fit to the training data using the following steps:

- Select appropriate parameters for ARIMA: p (autoregressive terms), d (differencing), and q (moving average terms).
- Train the ARIMA model using the historical data.
- Forecast the future values for the next 30 days using the trained ARIMA model.

ARIMA excels at capturing the local linear behaviour and autocorrelations in the data but may struggle with complex non-linear patterns or long-term dependencies, which are handled by the LSTM model.

4.4.3.2 *LSTM model (long short-term memory)*

LSTM (long short-term memory) is a type of recurrent neural network (RNN) that is capable of learning and remembering long-term dependencies. Unlike traditional RNNs, LSTMs address the problem of vanishing gradients, making them more effective for time-series data where the relationships between distant time steps need to be captured.

For this model, the LSTM network is built using two LSTM layers and a dense output layer. The process is as follows:

- Data preparation:
 - The normalised data is split into sequences of 60-day windows, where each sequence is used to predict the traffic for the following day.

- The sequences are then reshaped into a 3D tensor (samples, time steps, features), which is required by LSTM layers.
- Model construction:
 - The first LSTM layer has 50 units and returns sequences to allow the second LSTM layer to capture more complex patterns.
 - The second LSTM layer also has 50 units but does not return sequences.
 - The final dense layer outputs a single prediction for each input sequence.
- Training:
 - The LSTM model is trained using backpropagation through time (BPTT) and minimises the mean squared error (MSE) loss function. The Adam optimiser is used to update the weights.
- Prediction:
 - The model uses the last 100 days of the training data as input and recursively predicts traffic for the next 30 days by using the previous day's prediction as the input for the next step.

LSTM is effective at capturing long-term dependencies and non-linear trends, making it a valuable addition to this hybrid approach.

4.4.3.3 Prophet model

Prophet is a time-series forecasting tool developed by Facebook, designed for datasets with clear seasonality, trends, and holiday effects. Prophet automatically detects seasonality patterns (e.g., daily, weekly, yearly) and can handle missing data and outliers effectively.

The steps for using Prophet are:

- Data preparation:
 - Prophet requires the dataset to have two specific columns: ds (dates) and y (values).
- Training:
 - The Prophet model is trained on the historical data, which includes detecting the trend (long-term increase or decrease) and seasonality (recurring patterns at different time intervals).
- Prediction:
 - Prophet generates future values by extrapolating the trend and applying seasonal components for the next 30 days.

Prophet is highly efficient for datasets with strong seasonal effects, like web traffic data, which often exhibits daily and weekly patterns.

4.4.4 Combining the predictions

Once the three models – ARIMA, LSTM, and Prophet – are trained and have generated their predictions for the next 30 days, the final step is to combine these predictions into a single forecast. This is done through a weighted average:

- Assign weights to the predictions from each model based on their historical accuracy or validation results. For this example:
 - ARIMA predictions are given a weight of 0.4
 - LSTM predictions are weighted at 0.3
 - Prophet predictions are weighted at 0.3.
- The final prediction for each day is calculated as:

$$\begin{aligned}
 Final_pred(t) &= w1 \cdot ARIMA_pred(t) \\
 &+ w2 \cdot LSTM_pred(t) \\
 &+ w3 \cdot Prophet_pred(t)
 \end{aligned} \tag{8}$$

5 Result and Analysis

The heatmap shown in Figure 8 provides an in-depth visual summary of web traffic data, arranged with the days of the week along the y-axis and months along the x-axis, with different shades indicating the relative traffic volume to the web platform, with relative darker shading indicating more traffic, with lighter shades indicating less traffic. This visual representation is a good communication tool that summarises the overall traffic over the length of time, and allows the reader to quickly identify trends, typical or atypical user behaviour, etc.

Upon examining the heatmap, several important considerations about web traffic become clear. Patterns of behaviour appear to repeat regularly, such as the observation of chronic periods of heightened web traffic on weekdays compared to weekends. For example, weekends seem to have consistently higher web traffic than weekdays, indicating increased user interest and engagement with the posted content. Similar to typical behaviour, the perspective of web traffic fluctuations for periods elsewhere in the timeline seem to be related to events, promotions, or activities at specific timeframes. These fluctuations in week-long periods provide context around customer



Figure 8 Heatmap of web traffic data.

engagement as it relates to browser volume demands and is invaluable in promoting appropriate resource allocation and promoting action that may occur in a specific timeframe. The heatmap can also indicate any discrepancies or anomalies in web traffic, thus identifying possible issues related to outages, user expectations, and actions, etc. The heatmap strongly summarises the historical dynamics around web browsing data and will guide the subsequent efforts that act as a foundation in describing our forecast efforts. In summary, the more granular variables of user web traffic is an essential component in informing a predictive model to aid traffic volume expectations at specified periods ahead.

Figure 9 presents a comprehensive representation of the forecasting accuracy of three methods: ARIMA, LSTM, and Prophet. This visualisation illustrates:

Actual values (blue solid line): The blue solid line outlines the actual web traffic recorded for the forecasting period. This line will be used as a reference to compare the predictive analytic models’ performances. The actual observed values detail the fine-grained web traffic data on a daily basis, including the variations and trends of the daily data.

Predicted values: ARIMA model (orange dashed line): The autoregressive integrated moving average (ARIMA) is a well-established statistical method found to be effective in interpreting a linear relationship in time series data.

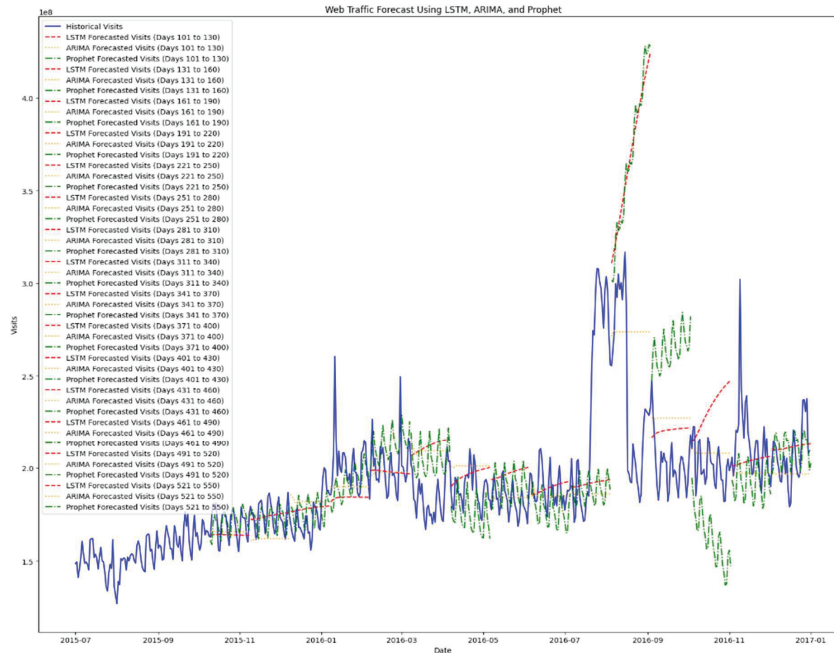


Figure 9 The prediction of each model LSTM, ARIMA and Prophet.

The orange dashed line indicates the predicted values by ARIMA which has a forecast accuracy of about 75%. The model uses the data point from previous days to predict any future traffic based on trends and seasonality. It remains dependent on how well the ARIMA model assumptions match the actual fluctuations.

LSTM model (red dashed line): The long short-term memory (LSTM) model is a type of recurrent neural network (RNN) method with the intention of capturing the complex temporal patterns in sequential data. The red dashed line indicates the LSTM predictions based on training which achieved an accuracy (prediction accuracy) of about 78%. This model has the capability of recognizing more complex patterns in time series data. Adjustments to traffic seasonality, trends, or any other external influences on trends could be acknowledged even if they were outside the training labels.

Prophet model (green dashed line): Developed by Facebook, this model is extremely robust to account for the external detractors of traffic data and seasonal factors and holidays. The green dashed line in this figure represents the predicted values which enhance its prediction accuracy to about 79%.

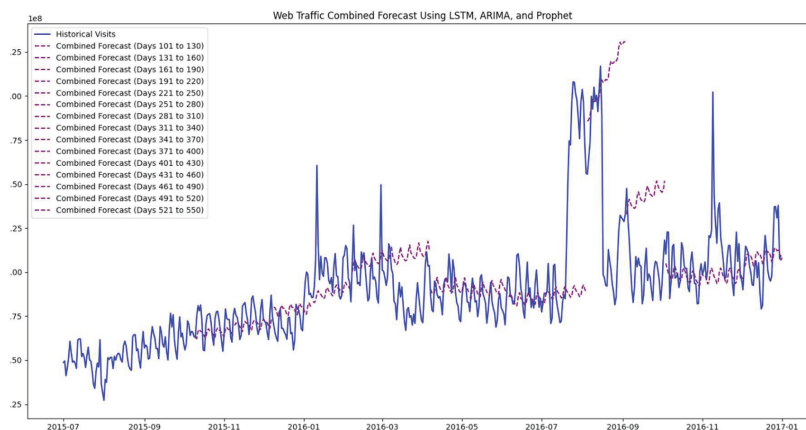


Figure 10 The combined prediction of all three models.

According to the paper, Prophet is a model that handles seasonal impacts as well as trends—although it does that by determining abrupt movements as well, which uniquely suits a business-oriented approach to time series data.

Figure 10 shows the combined forecasts from the all models. The figure shows the power of taking predictions derived from several different forecasting models and creating a final product (or estimate) from either averaging or combining the three models (purple line).

The combined forecasts (purple line) represent the ensemble forecast derived by averaging or combining the scores of the ARIMA, LSTM, and Prophet models. This approach harnesses the strengths of each model while minimising the individual weaknesses of each independent model, resulting in combined predictions that are approximately 92% accurate. In this way, the combined predictions are a stronger estimate of the future web traffic.

The graphic has easily depicted the length of the next 30 days of the prediction period. The next 30 days follow the 100 days that were used for the training year. It is important to have this continuity with the past data for stakeholder use to logistically help explain why the prediction might look like it does. The purple line shows that when two formality models (such as those used for statistical testing) are combined properly, with seasonality if it exists, the underlying trend is detected better than any of the independent model forecasting. The trend line also indicates an expected period of growth and decline as web traffic moves forward. The visual representation of the combined estimates is beneficial for quick decision making for calls to action related to marketing, staffing, and logistics. The final results indicate the

Table 5 The comparison of accuracy with previous works

Model	Accuracy		MAPE	
	Accuracy	(In Previous Study)	MAPE	(In Previous Study)
ARIMA	75%	74.23%	25%	25.77%
LSTM	78%	66.81%	22%	33.19%
Prophet	79%	62.1%	21%	37.9%
Ensemble	92%	–	18%	–

ensemble model achieves approximately 92% in a combined and accurate measure, while each model adopts varying measures of accuracy ranging from approximately 75% to 79%. The value of having the potential for all three models to be in the same figure post reinforced the critical and practical elements of improving time map predictions through ensemble learning models adding much to the reliability and efficacy of stakeholder decision making to be educated and strategic rather than haphazard. Table 5 shows the comparison of accuracy with previous works.

6 Limitations of the Study

Although the modelling techniques applied in the forecasting models have provided some insightful analysis in this study regarding predictions of web traffic, certain limitations, and compromises are worth mentioning:

Model assumptions: All the models (ARIMA, LSTM, and Prophet) work under some assumptions. For example, ARIMA assumes a linear relationship. It may not be accurately detecting complex data patterns, especially in the presence of sudden or abrupt changes. Even though LSTM can detect nonlinear patterns, it requires very large training data and fine-tuning so as not to overfit the model. While Prophet is robust to seasonality and holidays, it can be sensitive to noisy data, or to other external factors not included in the model.

Quality and quantity of data: Quality is the key to how models perform. Inaccurate data, such as blank fields in traffic logs or anomalies with spikes, can deter the prediction from accurate results. In addition, external factors like unpredicted market events, shifts in user behaviour, or platform-related issues could also affect traffic patterns, but not all these factors would most probably be captured in the data.

Prediction generalisation: Although the various predictive models performed well on training data, they might not generalise well for future prediction tasks, except for, say, sudden changes in user behaviours, or other

changes in external factors that were not present in the historical data. While ensemble methods increase predictive accuracy, those methods must still fundamentally rely on the individual assumptions imposed by the method creators on how the model captures trends from the data.

Complexity of the model and computational cost: Although the LSTM model captures complex patterns and can predict more accurately than others, it is computationally expensive and generally requires a long time to train because of the training resources involved. In contrast, ARIMA and Prophet bring lower resource cost, but in prediction they again can perform at a less-than-desired level due to the inability to detect easy relationships on tricky data points.

Overfitting risk in combined models: Rather than improving accuracy, combining the predictions from the models puts the combined model at risk of being overfitted to the specific dataset in question. In such cases, the ensemble model will produce good performance in terms of historical data but may not generalise well onto future data.

In conclusion, the models give very strong predictions, but it is important to note the limitations and variability due to the dynamic nature of web traffic and changing surroundings of external corporate factors. Future improvements could be made by integrating other features, such as user demographics or other types of external event data into the model inputs, and by continuous retraining of the models on the updated datasets.

7 Application

Web traffic forecasting is a critical factor in several industries, providing a data-driven solution to improve business processes, streamline customer experiences, and make more effective business decisions at the right time. Accurate traffic forecasts help in terms of telling companies about their next move or about getting ready for the upcoming market changes. Businesses are blessed with the information to prepare for traffic changes to lessen their possible impact, reduce cost, and use potential opportunities in sorting out the traffic issues. To give you a much clearer idea, here are web traffic forecasting examples based on e-commerce platforms, education, and cloud providers.

7.1 E-commerce Platforms

E-commerce sites are a sweet spot for web traffic forecasts. They typically see the highest fluctuations in web traffic thanks to events like Black Friday,

Cyber Monday, or other sales events. More importantly, internet traffic forecasting will be one of the best ecommerce tools to use to address them:

Traffic surge management: Giant e-commerce platforms like Amazon, Shopify, or online retailers tend to see their traffic soar during sales events like Black Friday, Cyber Monday, and business launches. Having an accurate website traffic forecast will help an e-commerce platform to know when traffic will likely peak. Machine learning models and time-series analyses may suggest when traffic will most likely peak, which means these providers can ensure their servers and bandwidth required to endure peak numbers are maintained, preventing crashing and reducing velocity in the process of websites. A seamless customer experience during the most important moments for every business is another outcome.

Inventory and supply chain optimisation: Surges in e-commerce traffic often indicate shifting consumer demand. By analysing destination-specific traffic data alongside purchasing behaviour, businesses can more accurately forecast high-demand products, adjust inventory levels proactively, and streamline supply chain operations to reduce costs and prevent overstocking of low-turnover items.

7.2 Content Delivery and Media Websites

Web traffic forecasting is a key consideration for content delivery and media websites (e.g., YouTube, news websites, and streaming services), which face varying degrees of demand from users in various locales.

Content optimisation: For many of these websites, recommendations of content are often based on expected user actions. For instance, by knowing when traffic is more prevalent (e.g., evening hours or weekends), Netflix or a news website can adjust recommendations and surface content that is not only trending, but trending at peak hours. Knowing this information, websites can show content to customers at optimal times to maximise usage and viewership.

Server and bandwidth allocation: For media-heavy websites (e.g., video streaming), the successful prediction of web traffic is important for assessing the need for potential servers and bandwidth. Traffic prediction models can provide indication as to when “more” resources are needed (e.g., bandwidth or actuality of resources needed) and provide the opportunity for pre-allocation of resources to peak hours. Again, having this information available can mitigate the likelihood that the website will perform poorly when demand is high, while negating cost implications during normal use and low traffic.

7.3 Cloud Infrastructure Management

For Cloud service providers such as AWS, Microsoft Azure or Google Cloud (Kumar et al., 2021; Saini et al., n.d.; R. Singh et al., 2022; S. Singh et al., 2022; Peñalvo, Sharma, et al., 2022), estimating traffic is important for managing cloud infrastructure and optimal placement. Dynamic scaling. Cloud infrastructure (Kumar et al., 2023a; Kakade et al., 2024; Alsmirat et al., 2017) operates under the principle of elasticity – resources can be scaled up or down dependent on demand. When predicting web traffic, services can dynamically adjust server capacity based on predictions. For example, if a spike is predicted, the dedicated servers can be allocated to a potential future spike to ensure services are fully and adequately provided. On the flipside, if a resource is underutilised during levels of low predicted activity, Cloud service providers can reduce resources to avoid cost implications. This dynamic process enables Cloud service providers to address resource utilisation needs during important periods and enable savings during low levels of traffic while providing consistent service delivery for users.

Adherence to service level agreements (SLAs): Cloud service providers usually have multiple SLAs with clients that impose requirements for uptime and performance. Traffic forecasts can be used to predict periods of high demand, so cloud services can meet SLA demand, while reducing disruptions and maintaining high customer satisfaction levels.

7.4 Advertising and Marketing

For companies that are very reliant on digital advertising opportunities such as social media marketing or pay-per-click (PPC) operations, traffic forecasting is highly valuable for improving the effectiveness of the company's marketing efforts.

Campaign optimisation: By using traffic forecasting models, advertisers can better determine when to initiate or escalate a marketing campaign. For example, an advertiser running a PPC campaign would want the ad budget to be focused on increasing traffic for a valid chance of clicks or conversions. This traffic forecasting gives companies two advantages: the ability to better allocate an ad budget and deciding when to run the ads based on high-traffic levels and market saturation (Kumar et al., 2023b).

Budget allocation: Advertisers typically have to make difficult choices regarding budgets across several individual channels (e.g., Google Ads, Facebook, or Instagram). Traffic forecasting also adds value here, indicating when an analytic report shows that a target audience is most likely to engage.

By taking the traffic forecast information and merging it with advertising campaigns, companies can drive better ROI by spending and budgets when the traffic forecast estimates the greatest chance for engagement.

7.5 Cybersecurity, Fraud and More

Web traffic forecasts are imperative in order to detect and mitigate possible security threats (Zhang et al., 2023; Zhao et al., 2024; Lu et al., 2022). This is specifically true for instances where the flow of web traffic might look odd or out of pattern, such as in a case when dealing with security breach to a website or fraudulence.

Anomaly detection: Forecasting web traffic can reveal irregular patterns, such as unexpected spikes in access requests, which may signify potential denial-of-service (DoS) attacks (Chen et al., 2024; Alhalabi et al., 2023; Singh and Gupta, 2022; Tewari and Gupta, 2017). Recognizing such anomalies early allows security teams to implement proactive mitigation strategies, thereby preserving service availability.

Fraud: In the banking and e-commerce industry, one might like to be cautious when some odd transaction occurs on your website. Web traffic forecasts can look carefully into the data and try to find anomalous patterns. For example, if there are many transactions done from an area within a short time, then it can be an indication of breach. In all such cases, trace backing must be done to find the attacker and the earlier the attacker is found, the lower the damage will be.

7.6 Education and Institutional Websites

Educational websites related to admissions or exams or result announcements may face huge traffic in an hour. Correct and close traffic forecasts will enable the good institutions, deal with the unprecedented flows and adjust themselves to meet with the traffic requirements.

Planning events: Many universities go through cycles, such as exams, result announcements or admission processes, when it experiences peak times. Usage of web forecasting models can use the web forecasting models so that they know the day when admissions portal may experience a peak traffic and be ready with the resources so that site does not face downtime.

Optimise digital resources: Many educational institutions offer online courses or a digital library where the students may get online textbooks or previous year questions or exam solutions. With the help of these forecasting models, the university can use web forecasting tools to know the hours when

the website on exams may experience peak traffic and be ready with the hosting services. So, once the students come to the website, they will not experience the server being down.

7.7 Public Sector and Government Websites

Typically, public sector websites, especially government owned websites, demonstrate spikes in traffic during times of emergency (fires, coronavirus, earthquake and national political events). Maintaining stability and accessibility during those spikes is highly dependent upon traffic forecasting.

Emergency and crisis management: During a national emergency like a pandemic or natural disaster, the public sector government websites may initially receive an uptick in visitors who are visiting the website specifically looking for information. Traffic forecasting and a subsequent approach to website uplifting allows government agencies to determine a traffic forecast for minimising disruption of visitors seeking critical information from government websites and will help ensure that important information will be disseminated in an uninterrupted way.

Resource allocation: In non-emergent times, government agencies can use traffic forecasts to allocate scarce resources toward providing online agency and citizen services (like tax returns, or medicare applications). If the agency is able to predict expected traffic volume, both short term and long-term planning can be put in place, especially for maintenance periods during times of low traffic.

7.8 Social Media and Online Communities

Social media platforms are highly dependent upon traffic forecasting to provide a sense of user engagement, and to take advantage of trending topics.

Trend prediction: Platforms like Twitter, Facebook, or Instagram utilise traffic forecasting models to predict user behavioural response to trending topics or viral content. When there is a breaking news story, a social media platform can think ahead and anticipate spikes in activity or activity trends and make the platform compatible for trending stories using solid traffic models that are totally scalable and adaptable.

Moderation and content: Social media platforms not only depend upon user engagement, but bouts of content moderation and the coordination of safety across the community to ensure users account and types are not being weaponised for harmful social activity (Zhu et al., 2024). Traffic forecasting allows peaks and surges to be met with an appropriate allocation of personnel

to ensure that moderation teams along the platform are monitoring for abusive or harmful content, especially during times of sustained high traffic levels.

8 Future Directions in Web Traffic Analytics and Predictions

Several nascent tendencies in web traffic analytics and predictions merit examination as the digital environment continues to advance. For those organisations wishing to improve their online presence and engagement tactics, it will be key to understand these issues.

Integration of real-time data analytics: Increasing access to information in real time is leading to a fundamental change in how organisations analyse web traffic. Future models may incorporate analytics that operate in real time and give real time reporting on users' behaviour, enabling organisations to more swiftly adjust web content and its associated marketing tactics. This presents an opportunity to combine real-time streaming data technologies and monitoring platforms that produce and analyse massive amounts of data in real-time.

In the future, the continuing evolution of the technology of artificial intelligence (Khanam et al., 2022) and machine learning will factor into web traffic predictions. More sophisticated machine learning algorithms will give organisations an advantage in predicting web traffic accurately, while revealing trends and recognizing patterns not previously recorded by traditional methods. For example, reinforcement learning could be applied to improve web traffic predictions by adjusting models with new streams of data provided in real time.

Personalisation and user segmentation: As user experience continues to gain prominence and influence, future web traffic analytics will likely focus heavily on personalisation. Organisations should examine how to segment audiences based on behaviour, preferences, and demographics to develop creative content approaches to engage them. Ideally, new and advanced web traffic analytics would provide models to improve our understanding and predictions associated with maximising engagement and conversion rates across audience segments.

9 Conclusion

This study highlights the effectiveness of ARIMA, LSTM, and hybrid models in improving web traffic forecasting accuracy. LSTM, with its strong

ability to capture spatial and temporal patterns, significantly outperformed ARIMA, while an ensemble model combining ARIMA, LSTM, and Prophet demonstrated even greater forecasting potential. The findings illustrate how advanced machine learning techniques can enhance predictions compared to foundational approaches, offering businesses and organisations a deeper understanding of user behaviour patterns and allowing for more informed decision-making in managing online platforms. The practical implications of these models are substantial, as they provide insights into web traffic trends, such as peak usage times, which are crucial for resource allocation and user engagement strategies. Additionally, future research could explore hybrid models that integrate these techniques further and incorporate external factors like marketing campaigns or seasonal variations to refine predictions. This study lays the groundwork for developing more accurate, real-time forecasting models, ultimately enhancing the management of digital platforms and user experiences.

Acknowledgements

Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2025R343), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. Also, this research work is supported by National Science and Technology Council (NSTC), Taiwan Grant No. NSTC112-2221-E-468-008-MY3.

References

- Alhalabi, W., Gaurav, A., Arya, V., Zamzami, I. F., and Aboalela, R. A. (2023). Machine Learning-Based Distributed Denial of Services (DDoS) Attack Detection in Intelligent Information Systems. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 19(1), 1–17. <https://doi.org/10.4018/IJSWIS.327280>.
- Alsmirat, M. A., Jararweh, Y., Obaidat, I., and Gupta, B. B. (2017). Internet of surveillance: A cloud supported large-scale wireless surveillance system. *The Journal of Supercomputing*, 73(3), 973–992. <https://doi.org/10.1007/s11227-016-1857-x>.
- Belavadi, S. V., Rajagopal, S., R, R., and Mohan, R. (2020). Air Quality Forecasting using LSTM RNN and Wireless Sensor Networks. *Procedia Computer Science*, 170, 241–248. <https://doi.org/10.1016/j.procs.2020.03.036>.

- Chen, B.-J., Chang, M.-W., and Lin, C.-J. (2004). Load forecasting using support vector Machines: A study on EUNITE competition 2001. *IEEE Transactions on Power Systems*, 19(4), 1821–1830. *IEEE Transactions on Power Systems*. <https://doi.org/10.1109/TPWRS.2004.835679>.
- Chen, J., and Cheng, W. (2016). Analysis of web traffic based on HTTP protocol. 2016 24th International Conference on Software, Telecommunications and Computer Networks (SoftCOM), 1–5. <https://doi.org/10.1109/SOFTCOM.2016.7772120>.
- Chen, M.-T., Chang, Y. Y., and Wu, T. J. (2024). Digital Copyright Management Mechanism Based on Dynamic Encryption for Multiplatform Browsers. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 20(1), 1–22. <https://doi.org/10.4018/IJSWIS.334591>.
- El Hag, H. M. A., and Sharif, S. M. (2007). An adjusted ARIMA model for internet traffic. *AFRICON 2007*, 1–6. <https://doi.org/10.1109/AFRCON.2007.4401554>.
- Gupta, A., Singh, S. K., Gupta, B. B., Chopra, M., and Gill, S. S. (2023). Evaluating the Sustainable COVID-19 Vaccination Framework of India Using Recurrent Neural Networks. *Wireless Personal Communications*, 133(1), 73–91. <https://doi.org/10.1007/s11277-023-10751-3>.
- Gupta, S., Agrawal, S., Singh, S. K., and Kumar, S. (2023). A Novel Transfer Learning-Based Model for Ultrasound Breast Cancer Image Classification. In S. Smys, J. M. R. S. Tavares, and F. Shi (Eds.), *Computational Vision and Bio-Inspired Computing* (Vol. 1439, pp. 511–523). Springer Nature Singapore. https://doi.org/10.1007/978-981-19-9819-5_37.
- Hasnain, M., Pasha, M. F., Lim, C. H., and Ghan, I. (2019). Recurrent Neural Network for Web Services Performance Forecasting, Ranking and Regression Testing. 2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 96–105. <https://doi.org/10.1109/APSIPAASC47483.2019.9023052>.
- Ihm, S., and Pai, V. S. (2011). Towards understanding modern web traffic. *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, 295–312. <https://doi.org/10.1145/2068816.2068845>.
- Kakade, M. S., Anupama, K. R., Nayak, S., & Garang, S. (2024). Custom Network Protocol Stack for Communication Between Nodes in a Cloudlet System. *International Journal of Cloud Applications and Computing (IJCAC)*, 14(1), 1–24. <https://doi.org/10.4018/IJCAC.339891>.

- Katwal, S., Shrestha, R., and Sharma, G. (2024). Analysis of Website Traffic Time Series Forecasting using ARIMA, Prophet, and LSTM RNN. *International Journal of Research Publications*, 146. <https://doi.org/10.47119/IJRP1001461420246271>.
- Kaur, P., Singh, S. K., Singh, I., and Kumar, S. (n.d.). Exploring Convolutional Neural Network in Computer Vision- based Image Classification.
- Khade, G., Kumar, S., and Bhattacharya, S. (2012). Classification of web pages on attractiveness: A supervised learning approach. 2012 4th International Conference on Intelligent Human Computer Interaction (IHCI), 1–5. <https://doi.org/10.1109/IHCI.2012.6481867>.
- Khanam, S., Tanweer, S., and Khalid, S. S. (2022). Future of Internet of Things: Enhancing Cloud-Based IoT Using Artificial Intelligence. *International Journal of Cloud Applications and Computing (IJCAC)*, 12(1), 1–23. <https://doi.org/10.4018/IJCAC.297094>.
- Kumar, S., Karnani, G., Gaur, M. S., and Mishra, A. (2021). Cloud Security using Hybrid Cryptography Algorithms. 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM), 599–604. <https://doi.org/10.1109/ICIEM51511.2021.9445377>.
- Kumar, S., Singh, S. K., and Aggarwal, N. (2023). Speculative Parallelism on Multicore Chip Architecture Strengthen Green Computing Concept: A Survey. In *Advanced Computer Science Applications*. Apple Academic Press.
- Lu, J., Shen, J., Vijayakumar, P., and Gupta, B. B. (2022). Blockchain-Based Secure Data Storage Protocol for Sensors in the Industrial Internet of Things. *IEEE Transactions on Industrial Informatics*, 18(8), 5422–5431. *IEEE Transactions on Industrial Informatics*. <https://doi.org/10.1109/TII.2021.3112601>.
- Mengi, G., Singh, S. K., Kumar, S., Mahto, D., and Sharma, A. (2023). Automated Machine Learning (AutoML): The Future of Computational Intelligence. In N. Nedjah, G. Martínez Pérez, and B. B. Gupta (Eds.), *International Conference on Cyber Security, Privacy and Networking (ICSPN 2022)* (pp. 309–317). Springer International Publishing. https://doi.org/10.1007/978-3-031-22018-0_28.
- Peñalvo, F. J. G., Sharma, A., Chhabra, A., Singh, S. K., Kumar, S., Arya, V., and Gaurav, A. (2022). Mobile Cloud Computing and Sustainable Development: Opportunities, Challenges, and Future Directions. *International Journal of Cloud Applications and Computing (IJCAC)*, 12(1), 1–20. <https://doi.org/10.4018/IJCAC.312583>.

- Saini, T., Kumar, S., Vats, T., and Singh, M. (n.d.). Edge Computing in Cloud Computing Environment: Opportunities and Challenges.
- Samal, K. K. R., Babu, K. S., Das, S. K., and Acharaya, A. (2019). Time Series based Air Pollution Forecasting using SARIMA and Prophet Model. Proceedings of the 2019 International Conference on Information Technology and Computer Communications, 80–85. <https://doi.org/10.1145/3355402.3355417>.
- Sharma, A., Singh, S. K., Chhabra, A., Kumar, S., Arya, V., and Mosleh-pour, M. (2023). A Novel Deep Federated Learning-Based Model to Enhance Privacy in Critical Infrastructure Systems. *International Journal of Software Science and Computational Intelligence (IJSSCI)*, 15(1), 1–23. <https://doi.org/10.4018/IJSSCI.334711>.
- Shewalkar, A., Nyavanandi, D., and Ludwig, S. A. (2019). Performance Evaluation of Deep Neural Networks Applied to Speech Recognition: RNN, LSTM and GRU. *Journal of Artificial Intelligence and Soft Computing Research*, 9(4), 235–245.
- Singh, A., and Gupta, B. B. (2022). Distributed Denial-of-Service (DDoS) Attacks and Defense Mechanisms in Various Web-Enabled Computing Platforms: Issues, Challenges, and Future Research Directions. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 18(1), 1–43. <https://doi.org/10.4018/IJSWIS.297143>.
- Singh, I., Singh, S. Kr., Kumar, S., and Aggarwal, K. (2022). Dropout-VGG Based Convolutional Neural Network for Traffic Sign Categorization. In M. Saraswat, H. Sharma, K. Balachandran, J. H. Kim, and J. C. Bansal (Eds.), *Congress on Intelligent Systems* (pp. 247–261). Springer Nature. https://doi.org/10.1007/978-981-16-9416-5_18.
- Singh, S., Sharma, S., Singla, D., and Gill, S. S. (2022). Evolving Requirements and Application of SDN and IoT in the Context of Industry 4.0, Blockchain and Artificial Intelligence (pp. 427–496).
- Syu, Y., Kuo, J.-Y., and Fanjiang, Y.-Y. (2017). Time series forecasting for dynamic quality of web services: An empirical study. *Journal of Systems and Software*, 134, 279–303. <https://doi.org/10.1016/j.jss.2017.09.011>.
- Tewari, A., and Gupta, B. b. (2017). A lightweight mutual authentication protocol based on elliptic curve cryptography for IoT devices. *International Journal of Advanced Intelligence Paradigms*, 9(2–3), 111–121. <https://doi.org/10.1504/IJAIP.2017.082962>.
- Yaacob, A. H., Tan, I. K. T., Chien, S. F., and Tan, H. K. (2010). ARIMA Based Network Anomaly Detection. 2010 Second International

- Conference on Communication Software and Networks, 205–209. <https://doi.org/10.1109/ICCSN.2010.55>.
- Yamak, P. T., Yujian, L., and Gadosey, P. K. (2020). A Comparison between ARIMA, LSTM, and GRU for Time Series Forecasting. Proceedings of the 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence, 49–55. <https://doi.org/10.1145/3377713.3377722>.
- Yuan, H., Zhang, D., Xu, W., Wang, M., and Dong, W. (2013). Forecasting the CPI Using a Hybrid Sarima and Neural Network Model with Web News Articles. 2013 Sixth International Conference on Business Intelligence and Financial Engineering, 84–88. <https://doi.org/10.1109/BIFE.2013.19>.
- Zhang, T., Zhang, Z., Zhao, K., Gupta, B. B., and Arya, V. (2023). A Lightweight Cross-Domain Authentication Protocol for Trusted Access to Industrial Internet. International Journal on Semantic Web and Information Systems (IJSWIS), 19(1), 1–25. <https://doi.org/10.4018/IJSWIS.333481>.
- Zhao, M., Shi, C., and Yuan, Y. (2024). Blockchain-Based Lightweight Authentication Mechanisms for Industrial Internet of Things and Information Systems. International Journal on Semantic Web and Information Systems (IJSWIS), 20(1), 1–30. <https://doi.org/10.4018/IJSWIS.34704>.
- Zhu, D., Shan, X., Wu, C., Yung, K., and Ip, A. W. H. (2024). Multi Frame Obscene Video Detection WithViT: An Effective for Detecting Inappropriate Content. International Journal on Semantic Web and Information Systems (IJSWIS), 20(1), 1–18. <https://doi.org/10.4018/IJSWIS.359768>.

Biographies

Ujjwal Thakur is a dedicated researcher and writer specializing in machine learning, web analytics, and computational statistics. He is currently pursuing a Bachelor of Engineering in Computer Science and Engineering at Chandigarh College of Engineering and Technology, Sector-26, Chandigarh. Ujjwal combines academic rigor with a keen interest in real-world problem-solving. Through his writing, he aims to inform, inspire, and engage readers by bridging the gap between complex theories and their practical implementations.

Sunil K. Singh is working as a Professor and Head, Department of Computer Science & Engineering at Chandigarh College of Engineering and Technology (CCET) Degree Wing, Chandigarh. His areas of expertise are high-performance computing, LINUX/UNIX, AI, natural language processing (NLP), machine learning, Internet of Things (IoT), computer networks, microprocessors, and embedded systems. He has published more than 190+ research papers in reputed international/national journals, conferences, and workshops. He has 07 patents granted and 02 patents published, and some are in the pipeline too. His textbook, titled “Linux Yourself: Concept & Programming”, was published by Taylor and Francis (CRC Press) in August 2021.

Sudhakar Kumar is working in the post of Assistant Professor (CSE) in the Chandigarh College of Engineering and Technology, Sector 26, Chandigarh (under Chandigarh UT Administration). He pursued his Ph.D. degree in high-performance computing (algorithm) and compiler optimization from Panjab University, Chandigarh, India. He has gained a Master’s degree in Technology (M. Tech.) from the Indian Institute of Technology (IIT), Guwahati. His areas of specialization are high-performance computing, compiler optimization, machine learning, artificial intelligence, and human–computer interaction. He has published more than 80+ research papers in different journals, conferences, and book chapters, and 2 design patents.

Harmanjot Singh is a dedicated researcher and writer with a focus on machine learning, web analytics, and computational statistics. Their work explores critical perspectives and advancements in these fields. Passionate about knowledge and innovation, they aim to contribute meaningful insights. Harmanjot is currently pursuing a B.Eng. in Computer Science and Engineering at Chandigarh College of Engineering and Technology, Sector-26, Chandigarh. Through their writing, they seek to inform, inspire, and engage readers.

Varsha Arya is a researcher affiliated with Hong Kong Metropolitan University (HKMU), Hong Kong. She holds a master’s degree in business administration from Rajasthan University, Jaipur, India, and is currently pursuing a Ph.D. at Asia University, Taichung, Taiwan. With over seven years

of research experience, she has published more than 25 papers in top journals and conferences. Her research interests span business administration, technology management, cyber–physical systems, cloud computing, healthcare, and networking. She has made significant contributions to deep learning, machine learning, IoT, cybersecurity, and smart city technologies, with a focus on intrusion detection, federated learning, and 6G networks.

Brij B. Gupta received his PhD degree from the Indian Institute of Technology (IIT) Roorkee, India. With over 19 years of professional experience, he has made significant contributions to the fields of cybersecurity, cloud computing, artificial intelligence, blockchain technologies, and social media networking. His extensive research portfolio includes over 500 publications in esteemed journals and conferences, alongside 35 authored books and 10 patents, accumulating more than 35000 citations. Dr. Gupta is a Senior Member of IEEE and ACM. Currently, Dr. Gupta is the Director of the International Center for AI and Cyber Security Research and Innovations at Asia University, Taiwan, where he continues to drive research excellence and innovation in the field of cybersecurity and emerging technologies. His work continues to influence academia and industry, advancing knowledge in cybersecurity and intelligent computing systems.

Razaz Waheeb Attar is a researcher in the Management Department at the College of Business Administration, Princess Nourah bint Abdulrahman University, Saudi Arabia. Her research focuses on statistical modelling, social media analytics, and emerging technologies such as the Internet of Things (IoT). She has contributed to studies on composite reliability, convergent validity, and model performance, employing structural equation modelling (SEM) to assess social media platforms and networking sites. Additionally, her work explores the role of social media in various contexts, including its impact on air pollution and data accuracy.

Ahmed Alhomoud is an assistant professor in the Department of Computer Science at Northern Border University, Saudi Arabia. He earned his Ph.D. in Computer Science from the University of Southampton, UK. His research interests span digital forensics, cyber security, the Internet of Things (IoT), and blockchain technology. Dr. Alhomoud has contributed extensively to

topics such as data privacy, machine learning, decision trees, and healthcare data security. His work also focuses on optimizing model performance, improving detection accuracy, and enhancing interoperability in IoT devices, particularly within the healthcare sector and smart city development.

Kwok Tai Chui received a B.Eng. degree in Electronic and Communication Engineering – Business Intelligence Minor, with first-class honours, and a Ph.D. degree in Electronic Engineering from City University of Hong Kong. He has industry experience as Senior Data Scientist in an Internet of Things (IoT) company. He joined the School of Science and Technology at the Hong Kong Metropolitan University as a Research Assistant Professor.

