TIME SERIES ANALYSIS METHODS IN TRACKING USER ACTIVITY: FROM EVENT COLLECTION TO BEHAVIOR MODELING

Kuznetcov I.
bachelor's degree, Udmurt State University
https://doi.org/10.5281/zenodo.14499294

Abstract

This article explores the methods of time series analysis used to track user activity, focusing on event collection, preprocessing, and the modeling of user behavior. The paper examines traditional statistical methods like AutoRegressive Integrated Moving Average (ARIMA) and exponential smoothing, as well as modern machine learning approaches such as Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU). The article also discusses the challenges of working with user data and demonstrates how it can be utilized to predict and understand audience behavior patterns.

Keywords: time series, user activity analysis, machine learning, AutoRegressive Integrated Moving Average (ARIMA), Seasonal-Trend decomposition using Loess (STL), Recurrent Neural Networks (RNN).

Introduction

In recent years, user activity analysis has become an integral part of research and commercial tasks, such as predicting consumer preferences, personalizing content, and optimizing user interactions. One of the most effective methods for studying user behavior is the use of time series analysis. Time series enable modeling of user activity dynamics, as well as identify patterns and anomalies, which provides valuable information for business and research.

The relevance of time series analysis in tracking user activity is due to the increasing volume of data and its irregular nature. There is a need to develop methods that would take into account not only the regularity and trends of data, but also the unpredictable changes in user behavior, as well as seasonal fluctuations typical for many digital platforms. In this context, it is important to highlight both traditional statistical methods and new approaches, such as machine learning, which can significantly improve the accuracy of the analysis.

The purpose of this paper is to explore existing methods of time series analysis, their capabilities and limitations in the context of tracking user activity. The article will also consider the impact of data quality on the results of the analysis and offer recommendations for improving time series processing methods.

Main part. Data collection and preprocessing

To effectively analyze user activity time series, proper organization of the data at the **collection stage** is of primary importance. Data collected from users typically consists of events recorded at varying frequency and structure. These may include user actions on a web page, clicks, time spent on a page, transitions between sections, as well as more complex interactions such as transactions or video views. It is important that the process of collecting this data is planned with the need for subsequent processing and analysis in mind [1].

One of the main methods of collecting data is the use of **event tracking systems**. Systems such as Google Analytics or more specialized tools such as Mixpanel or Segment allow you to collect information about user actions in real time. However, when using such systems, it is important to consider the limitations of data accuracy and completeness. For example, there are often situations when events are not recorded due to technical failures or limitations on the client side, such as JavaScript problems or cookie disabling.

Another approach is to **collect data via API**, which is especially relevant for mobile applications and services that work with large volumes of information. Here, data on user actions can be aggregated and sent as a sequence of events in real time. However, it is important to consider possible problems with delays in data transmission, which can lead to discrepancies between the user's actual activity and the registered events.

The next step is **preprocessing the collected data**. Time series obtained from various sources often require significant effort to prepare them for further analysis. One of the first tasks is to remove or process gaps in the data. Gaps can occur for various reasons: the user may close the application or browser before the event is recorded, or there may be technical failures. In such cases, various methods for filling gaps are used, such as linear interpolation or smoothing methods, if they do not distort the dynamics of the time series [2].

It is also important to **filter out noise data**. In real-world conditions, collected data often contains random outliers that can significantly distort the results of the analysis. To remove them, anomaly detection methods based on statistical approaches or machine learning models are used. For example, using outlier detection algorithms such as the moving average method or the Gaussian distribution method allows you to effectively filter out atypical values.

An equally important task is **data normalization**. Time series can be presented in different units of measurement (e.g. number of clicks, time in seconds, number of views), which makes their subsequent analysis difficult. To bring data to a single scale, normalization by standard deviation or scaling is used, when the data is brought to the range [0,1]. This eliminates the scaling effect, which can distort the results of the analysis [3].

In addition, at this stage it is also important to pay attention to the alignment of time series, especially if the data is collected from several sources. For example, if events are recorded at different frequencies (some – every second, others – every minute), then it is necessary to bring them to one time grid for correct analysis.

This can be done using respacing methods or aggregating data at a higher level, for example, by hours or days.

Data preparation is one of the most labor-intensive stages in time series analysis, since the accuracy and reliability of subsequent conclusions depend on the quality of the collected and processed data. At this stage, it is extremely important to take into account all the nuances associated with the quality of the data, its completeness and regularity in order to ensure the reliability of further modeling of user behavior.

Time series analysis methods

Time series analysis methods play an important role in identifying patterns and predicting user behavior. Depending on the complexity of the data and the goals of the analysis, different approaches are chosen, including both traditional statistical methods and modern models based on machine learning (table 1).

Table 1.

Time series analysis techniques [4, 5]

Analysis method	Description	Advantages
ARIMA (AutoRegressive Integrated Moving Average)	Modeling of a time series using autoregression (AR) and moving averages (MA), suitable for stationary data.	Simple to apply, suitable for linear trends and seasonality.
Exponential smoothing	Forecasting based on weighted averages of previous values, with weights decreasing as time moves further from the present.	Suitable for short-term forecasting, effective for regular data patterns.
STL (Seasonal-Trend decomposition using Loess)	Decomposition of a time series into components: trend, seasonality, and residuals.	Excellent for data with clear seasonality and trends.
RNN (Recurrent Neural Networks)	Neural networks are capable of considering dependencies on previous values in a sequence.	Ideal for time-dependent data, but may suffer from the vanishing gradient problem.
LSTM (Long Short-Term Memory)	An extension of RNN that includes memory cells to retain long-term dependencies.	Handles long-term dependencies well and addresses the vanishing gradient problem.
GRU (Gated Recurrent Unit)	A simplified version of LSTM with two primary gates: update and reset.	Faster to train and requires fewer computational resources than LSTM, good for time-limited tasks.

Thus, time series analysis methods play a key role in tracking user activity, allowing not only to predict future events but also to identify hidden patterns that can form the basis for improving user experience and optimizing processes on digital platforms.

Interpretation of results

Once time series analysis methods have been applied to track user activity, the next step is to interpret the results. This involves identifying key trends, such as seasonal changes in activity, as well as assessing anomalies that may signal technical issues or significant changes in user behavior. For example, a spike in visits could be the result of a website error or a viral effect on social media.

Seasonal fluctuations, such as increased traffic during holidays or on specific days of the week, play a crucial role in planning marketing strategies. For their analysis, it is essential to account not only for trends but also for recurring patterns, as well as to distinguish between random fluctuations and long-term changes in user preferences.

Specialized time series models, such as ARIMA, its extension SARIMA (Seasonal ARIMA) for capturing seasonality, and the Holt-Winters method, which effectively handles trends and seasonal effects, can be employed for forecasting such data. These approaches enhance the accuracy of the analysis, account for temporal data features, and enable more efficient resource planning. Research has shown that for short-term forecasting of mobile traffic, the SARIMA approaches and the Holt-Winters (HS) model provide high accuracy with good results for various error estimates (fig. 1).

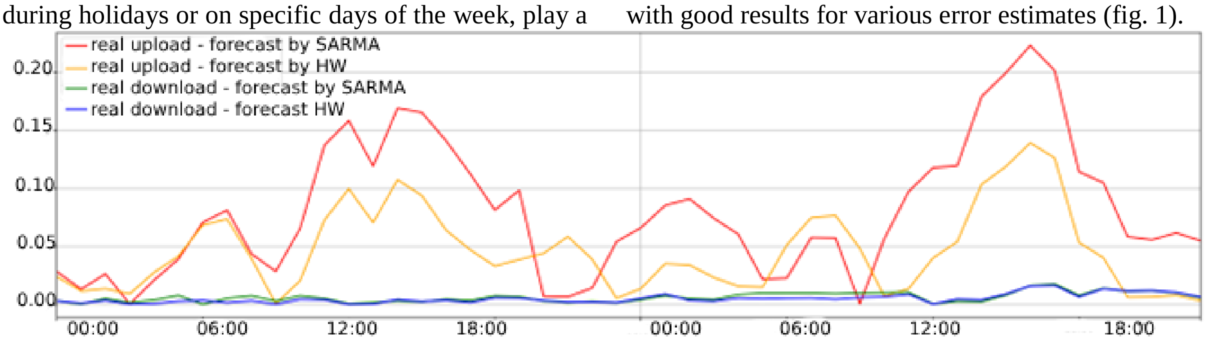

Figure 1. Short-term forecasting results, absolute error [7]

It is also important to consider the time period for which the data is collected. Long-term data allows you to identify stable trends, while short-term data may reflect random fluctuations. The influence of external factors (such as economic crises or marketing campaigns) should also be taken into account when interpreting the data to obtain more accurate conclusions.

Assessing the accuracy of a model is an important step in interpreting the results of time series analysis. Two of the most common metrics for this task are **MSE** and **MAE**. MSE measures the average squared deviation of predicted values from actual ones, which helps to identify large errors in forecasts. MAE, in turn, provides a more intuitive idea of the average deviation, measuring the difference between actual and predicted values without taking into account the direction of the error.

Additionally, the root **mean square error** (RMSE), which is the square root of the MSE and helps interpret the error in the same units as the data, is often used to more accurately evaluate the model. The **coefficient of determinatio**n (R^2) is another important metric, showing the proportion of the variance of the dependent variable explained by the model. High values R^2indicate that the model predicts the data well, while low values indicate a weak relationship between the forecast and the actual data. It is important to remember that overfitting the model, in which the model adjusts too closely to historical data, can lead to poor results on new data. It is important to regularly validate the model on different subsamples of the data to minimize this risk.

Finally, it is important to consider the interpretability of models when interpreting results, especially when using complex machine learning algorithms such as neural networks. In such cases, it is necessary to apply methods that can explain the model's output, such as feature importance analysis or visualization techniques. This helps to understand which factors had the greatest impact on the predicted events and facilitates data-driven decision making. For example, **Recora's** app found that users were struggling with button-related functionality. Using behavioral analytics tools such as heatmaps and session replays, the team found that users were incorrectly interacting with the button, expecting it to take a long press instead of a short press. A simple fix for this issue reduced support requests by 142% [8].

Another example is **Inspire Fitness**. To improve their online fitness course platform, the company used analytics data to identify which classes and instructors were most popular with users. This data allowed them to develop more engaging content and increase app usage by 460% [9].

Conclusion

Time series analysis is a powerful tool for tracking and predicting user activity, providing important information to improve user experience and optimize processes on digital platforms. The use of methods such as ARIMA, exponential smoothing, STL, and machine learning models such as RNN, LSTM, and GRU allows you to effectively identify trends and anomalies in user behavior, predict their actions, and adapt the strategy in real time. However, the choice of analysis method depends on the specifics of the data and the objectives of the study, as well as the quality and completeness of the data collected.

An important aspect is also the interpretation of the results, which includes identifying hidden patterns and anomalies, as well as using appropriate metrics to evaluate the accuracy of the models. Models focused on time series analysis must take into account many factors, including seasonal fluctuations, external influences, and user behavior. It is also important to remember the interpretability of complex models, which facilitates more informed decision-making based on the obtained data. Using these methods gives companies and researchers the opportunity to more accurately predict user behavior and create personalized recommendations to improve the user experience.

References:

1. Parashar A., Parashar A., Ding W., Shabaz M., Rida I. Data preprocessing and feature selection techniques in gait recognition: A comparative study of machine learning and deep learning approaches //Pattern Recognition Letters. 2023. Vol. 172. P. 65-73.
2. Tawakuli A. Survey: time-series data preprocessing: a survey and an empirical analysis. // Journal of Engineering Research, 2024.
3. Liu Z. Adaptive normalization for non-stationary time series forecasting: A temporal slice perspective. // Advances in Neural Information Processing Systems, 2024. Main Conference Track.
4. Mironov A. Uneven time series forecasting using a modified exponential smoothing method. // In E3S Web of Conferences, 2024. Vol. 531. P. 03-20.
5. Khaldi R. What is the best RNN-cell structure to forecast each time series behavior? // Expert Systems with Applications, 2024. Vol. 215. P.119-140.
6. Ponomarev E.V. Using Artificial Intelligence to enhance user experience in mobile applications // Innovacionnaya nauka. 2024. No. 9-2. P. 54-59.
7. Kochetkova I. Short-term mobile network traffic forecasting using seasonal ARIMA and holt-winters models. // Future Internet, 2023. Vol. 15. No. 290. P. 01-15
8. Case Study Recora / UXcam // URL: https://uxcam.com/case-study/recora/ (date of application: 22.10.2024).
9. Case Study Inspire Fitness / UXcam // URL: https://uxcam.com/case-study/inspire-fitness/ (date of application: 25.10.2024).