IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 1 April 2024, accepted 12 May 2024, date of publication 16 May 2024, date of current version 23 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3402133

RESEARCH ARTICLE

Sales Forecasting of Overrated Products: Fine Tuning of Customer's Rating by Integrating Sentiment Analysis

PARTHA GHOSH[1], OENDRILA SAMANTA[1], TAKAAKI GOTO[2], (Member, IEEE), AND SOUMYA SEN[3], (Member, IEEE)

[1]Department of Computer Science and Engineering, Academy of Technology, Aedconagar Hooghly, West Bengal 712121, India
[2]Faculty of Information Sciences and Arts, Toyo University, Kawagoe 350-8585, Japan
[3]A. K. Choudhury School of Information Technology, University of Calcutta, Kolkata 700106, India

Corresponding authors: Oendrila Samanta (oendrila.payel@gmail.com) and Partha Ghosh (pghosh44@gmail.com)

This work was supported by Toyo University, Japan.

ABSTRACT Enhancement of the profitability of any business organization is driven by proper forecasting. However, this is challenging as many factors affect the forecasting and the degree of relevant factors changes over time. Henceforth, it is essential for any business organization to develop a reliable and consistent sales forecasting model that can drive their growth. In today's business environment, customer ratings play a pivotal role in evaluating business performance, particularly in online retailing. These ratings provide valuable insights into the strengths and weaknesses of a product or service. The rating values are generally a set of integer values within a given range. This policy restricts users from expressing their views as they may wish to give a value that is not an integer. Hence, the system fails to capture the actual view of the customer about a certain product or service. As the intermediate values (decimal values) are not permitted, customers are generally compelled to round up their ratings, resulting overrating products. This problem can be addressed if textual reviews from the customers are recorded and these are analyzed for judging customers' satisfaction level. In this research work, we compute customer satisfaction by analyzing the review text of each customer for a particular product by using VADER sentiment analysis tool and use this result for tuning the actual user given ratings. A novel model is proposed to consider the tuned average customer rating amalgamating with standard forecasting methods like ARIMA, SARIMA, and LSTM. The experimental results on the Amazon dataset reveal 10% to 96% improvement in forecasted values for different types of products.

INDEX TERMS ARIMA, forecasting, LSTM, review, rating, SARIMA, sales, sentiment analysis, VADER.

I. INTRODUCTION

Establishing a harmonious relationship between production and market demand is paramount for sustained profit growth and market dominance, all while optimizing resource allocation. By aligning production output with consumer needs, companies can capitalize on opportunities, maximize revenue, and solidify their position in the market without overextending resources. Consequently, organizations across all sectors recognize the vital role of business forecasting [1], [2], [3], [4]. By utilizing a precise business forecasting [3], [5], [6], [7] model, companies can minimize risks, optimize production processes, effectively manage the supply chain [8], and make informed decisions regarding manpower and capacity planning. A well-suited forecasting model can propel a company towards success, whereas the lack thereof can impede progress. In today's dynamic and highly competitive business environment, the assessment and comprehension of business performance are significantly dependent on the pivotal role played by customer ratings [9].

The associate editor coordinating the review of this manuscript and approving it for publication was Akin Tascikaraoglu.

These evaluations, offered by the customers themselves, furnish valuable perspectives on diverse facets of product performance, encompassing product quality, customer service, satisfaction levels, and the overall reputation of the brand. As the business landscape continues to evolve rapidly [10], these customer ratings serve as indispensable tools for businesses to gauge their strengths and areas for improvement. The insights derived from these ratings empower companies to make informed decisions, refine their products or services, and enhance overall customer experience. In essence, customer ratings are integral in shaping the perception of a brand in the market and are instrumental in steering businesses toward sustained success amidst the challenges of a dynamic marketplace. By closely analyzing and interpreting customer ratings, businesses gain valuable knowledge about their strengths, weaknesses, and areas for improvement [3], [11], [12], enabling them to make data-driven decisions and implement targeted strategies to enhance their performance. With the advent of online platforms [13] and social media, customer ratings have gained even greater significance [8], [9], [11], as they can quickly reach a wide audience and significantly influence consumer perceptions and purchasing decisions.

But in most of the online retail platforms, customers often encounter limitations when it comes to rating products or services, as they are typically restricted to whole numbers within a predefined range [9]. This constraint poses a challenge because customers often desire to provide ratings that fall between consecutive integers, allowing for a more precise reflection of their satisfaction levels with a specific product or service. Regrettably, these online platforms do not permit the use of decimal values in customer ratings, thereby restricting the ability of customers to express their opinions accurately. As a result, customers often round up their ratings to the next whole number, leading to products being overrated. The cumulative impact of this prevalent practice can lead to a product being consistently overrated. Therefore, this limitation can lead to a certain degree of inaccuracy in the overall assessment of a product and that can result in inaccurate prediction of sales for that specific product. In this scenario, review text analysis [14] is indeed an alternative approach for assessing customers' sentiment regarding a product or service. Instead of relying solely on numerical ratings, analyzing the content of customer reviews provides valuable insights into their opinions, experiences, and sentiments. This qualitative analysis complements the quantitative ratings by delving deeper into the reasons behind customers' satisfaction or dissatisfaction.

Hence, to predict the sales of a specific product accurately, it's important to consider both customers' review text and their ratings as equally significant factors. In this paper, by extracting and analyzing the textual content of customer's review, we assign sentiment scores to individual reviews within the range from -1 to $+1$ based on the expressed opinions about a particular product. Where -1 signifies extreme negative sentiment and $+1$ signifies extreme positive sentiment respectively. Any number (integer or rational) in between signifies a mixture of positive and negative sentiments. These are further used for tuning the actual user rating. Thereafter we have shown that, with this tuned aggregated customer rating piecing together with standard forecasting models generating better forecasted results in case of overrated products.

The organization of this paper is as follows: Literature survey and the motivation behind this work are in section II. The selection of models and evaluation metrics are in section III. The proposed model is in section IV. We perform a case study in section V, followed by a result analysis in section VI. Finally, we conclude in section VII.

II. LITERATURE SURVEY AND MOTIVATION OF RESEARCH

For modern organizations, forecasting product sales [1], [2], [3], [12] is vital for business growth. Various time series forecasting models [1], [5], [12], [15] have been proposed to minimize disparities between predicted and actual growth, aiming to enhance accuracy and strategic planning for sustained success. An autoregressive (AR) component and summarized combination of drift and regression error based forecasting model ARIMA [16], [17] was proposed which assumes that the time series being modeled is linear and follows a specific structure. ARIMA model's goal is to determine the values of p (autoregressive order), d (Integrated order), and q (moving average order) that best fit with the time series data. This is often done through a process called model selection, where different combinations of p, d, and q are evaluated based on their ability to accurately predict future values. A modification is made on this model, applicable on the time series data that exhibits seasonality known as SARIMA [18], [19]. This model produces good result especially when patterns repeat at regular intervals. The Seasonal Autoregressive (SAR) value depicts the connection between an observation and its observations with a seasonal lag. Further an enhanced forecasting model LSTM [20], [21], [22], a type of RNN (Recurrent Neural Network) [23], was developed to preserve and preserve relationships within data sequences. It demonstrates outstanding performance in tasks such as time series prediction [24], [25], [26], [27], natural language processing, and speech recognition. Its key strength lies in its ability to proficiently capture and retain long-term dependencies, thereby enhancing its effectiveness across various domains [28], [29]. This capability is facilitated by a memory cell and controlled information flow gates.

Consequently, the success of a business or brand is intricately connected to the satisfaction of its customers [9], [30]. In instances where a customer's experience with a product is unfavorable, it becomes imperative to refine and enhance the product's functionality [30]. In online retailing, generally customers have the option to express their satisfaction about a product either by providing numerical ratings and/or expressing it in written textual format [31].

Hence, we can deduce that the true level of satisfaction is represented by the combined feedback of numerical ratings and written textual comments about that product. Consequently, to identify such level of satisfaction, analysis of customer sentiments are essential. This analytical process is formally known as Sentiment Analysis [14], [32], [33], [34]. Sentiment analysis involves the identification and categorization of opinions expressed both within textual content and through numerical ratings, ultimately leading to the determination of whether these sentiments are positive or negative in nature. In Gilbert's research [32], the development of VADER (Valence Aware Dictionary and sEntiment Reasoner) [32], [33], [34], a simplistic rule-based model for general sentiment analysis, was explored. The study evaluated VADER's performance by comparing it against 11 commonly used benchmarks [32], including ANEW, LIWC, the General Inquirer, Senti WordNet, and machine learning techniques like Naive Bayes, Maximum Entropy, and Support Vector Machine (SVM) [32]. The research encompassed VADER's creation, validation, and assessment stages, employing a blend of qualitative and quantitative methods to construct and validate a sentiment lexicon tailored for social media. VADER's notable strengths included its enhanced sensitivity to sentiment expressions in social media contexts and its ability to generalize effectively across diverse domains, setting it apart from traditional sentiment lexicons such as LIWC [32].

Taking all these considerations into account, the primary aim of this research is to delve into the correlation between sentiment values and their influence on the prospective sales of a product. The investigation will center on comprehending the extent to which emotional and attitudinal expressions, reflected in sentiment values, can serve as predictors for future sales trends. By scrutinizing customer sentiments, opinions, and feedback, the study seeks to unveil valuable insights into consumer behavior and preferences which in fact can become an important factor for future sales of that product.

III. METHODS AND MATERIALS

A. MODEL SELECTION AND DESCRIPTION

The proposed model is divided into two parts. In the first part, we tried to comprehend the sentiment of customers about a product and in the second part, using that sentiment along with review score we forecast the sales of that product.

In the first part, for sentiment analysis, we have employed the VADER [32], [33], [34] Sentiment Analysis tool. VADER represents a sentiment analysis tool that relies on both lexicon-based and rule-based techniques, designed with a particular sensitivity to sentiments expressed in online media. VADER operates by leveraging a combination of lexical features, primarily marked with their semantic orientation as either positive or negative. Consequently, VADER not only provides a polarity score but also offers insights into the degree of positivity or negativity associated with a sentiment.

VADER uses a very much well defined and well-organized lexicon for manage data from social media and the everyday conversations of individuals. As one might anticipate, lexicons can be both expensive and time-consuming to develop. Consequently, they tend to be infrequently updated, leading to a lack of the most current slang and expressions used to convey a person's emotions. Consider the Amazon review [35] analysis below as an example. In the following Amazon reviews [35], all the elements that indicate the person's unhappiness or happiness (bold symbols) are informal expressions, including numerous punctuation marks, abbreviations, and emojis. If this type of information is not taken into account, a sentiment analysis algorithm might incorrectly classify this review as neutral. VADER is well-equipped to handle such terms within its lexicon. For example, consider the following three reviews [35]:

Review 1: This is a great wallet. I would rate it 4 and a half stars. I like the comics print. If the wallet could be just a tiny little (tinny little **wtf??**) It would be perfect. But its a really nice wallet you can put money there if you were wondering.

Review 2: I just received my order today and the rings are awesome! They are a good weight not cheap and flimsy like I feared they'd be. They sparkle like crazy and dare I say they look like the real deal? **;)** I am very impressed and I can't wait to wear them. Thanks Amazon. **:)**

Review 3: The toe is not flat/level. The platform sandal version of this boot by the same company has a flat/level toe - PERFECT for dancing and walking. I hoped this boot would be the same, but it's not. **:(** Making them work, but disappointed.

In the second part of this research, we aimed to enhance sales forecasting by integrating this sentiment of customers along with their review score with well-known forecasting methods such as ARIMA, SARIMA, and LSTM [7]. The computation process of each model is elaborated upon in the following subsections.

1) ARIMA(p, d, q) MODEL

ARIMA, which stands for Autoregressive Integrated Moving Average, is designed to capture and predict the underlying patterns and trends in time series data. The Autoregressive (AR) component models the relationship between the current value in the time series (Y_t) and its past values ($Y_{(t-1)}$, $Y_{(t-2)}, \cdots, Y_{(t-p)}$), where '$p$' is the autoregressive order. Mathematically, it can be expressed as shown in Equation-1.

$$Y_t = c + \varphi_1 * Y_{(t-1)} + \varphi_2 * Y_{(t-2)} + \ldots + \varphi_p * Y_{(t-p)} + \varepsilon_t \tag{1}$$

Here, $\varphi_1, \varphi_2, \cdots, \varphi_p$ are the autoregressive coefficients, 'c' is a constant, and ε_t is the current white noise error term.

The Integrated (I) component represents the differencing order ('d'), which determines the number of times differencing is applied to the time series data to make it stationary. The differencing operation is denoted as 'Δ' and is calculated as shown in Equation-2.

$$\Delta\ Y_t = Y_t - Y_{(t-1)} \tag{2}$$

The operation as shown in Equation-2 is repeated 'd' times to achieve stationarity if needed. The Moving Average (MA) component models the relationship between the current value in the time series (Y_t) and past forecast errors or residuals ($\varepsilon_{(t-1)}, \varepsilon_{(t-2)}, \cdots, \varepsilon_{(t-q)}$), where '$q$' is the moving average order. Mathematically, it can be expressed as shown in Equation-3.

$$Y_t = c + \varepsilon_t + \theta_1 * \varepsilon_{(t-1)} + \theta_2 * \varepsilon_{(t-2)} + \ldots + \theta_q * \varepsilon_{(t-q)} \tag{3}$$

Here, $\theta_1, \theta_2, \cdots, \theta_q$ are the moving average coefficients, 'c' is a constant, and ε_t is the current white noise error term. Therefore, the full ARIMA(p, d, q) model is a combination of the AR, I, and MA components and can be expressed as shown in Equation-4.

$$\begin{aligned} Y_t = c &+ \varphi_1 * \Delta\ Y_{(t-1)} + \varphi_2 * \Delta\ Y_{(t-2)} + \cdots \\ &+ \varphi_p * \Delta\ Y_{(t-p)} + \varepsilon_t - \theta_1 * \varepsilon_{(t-1)} \\ &- \theta_2 * \varepsilon_{(t-2)} - \ldots - \theta_q * \varepsilon_{(t-q)} \end{aligned} \tag{4}$$

The autoregressive (φ) and moving average (θ) coefficients are estimated from historical data. Once the model is estimated and validated, it can be used for forecasting future values of the time series based on the estimated coefficients.

2) SARIMA MODEL

A Seasonal Autoregressive Integrated Moving Average (SARIMA) model is an extension of the ARIMA model that is designed to handle time series data with both non-seasonal and seasonal patterns. The mathematical equation for SARIMA is quite complex due to the combination of non-seasonal and seasonal components, the overall expression is shown in the Equation-5.

$$\begin{aligned} Y_t = c &+ \varphi_1 * \Delta\ Y_{(t-1)} + \varphi_2 * \Delta\ Y_{(t-2)} + \cdots \\ &+ \varphi_p * \Delta\ Y_{(t-p)} + \Phi_1 * \Delta\ Y_{(t-s)} + \Phi_2 * \Delta\ Y_{(t-2s)} + \cdots \\ &+ \Phi_P * \Delta\ Y_{(t-Ps)} - \theta_1 * \varepsilon_{(t-1)} - \theta_2 * \varepsilon_{(t-2)} - \ldots \\ &- \theta_q * \varepsilon_{(t-q)} - \Theta_1 * \varepsilon_{(t-s)} - \Theta_2 * \varepsilon_{(t-2s)} - \ldots \\ &- \Theta_Q * \varepsilon_{(t-Qs)} + \varepsilon_t \end{aligned} \tag{5}$$

where, $\Delta Y_{(t-1)}, \Delta Y_{(t-2)}, \cdots, \Delta Y_{(t-p)}$ are the non-seasonal differences at various lags, which are used in the non-seasonal autoregressive component. $\Delta Y_{(t-s)}, \Delta Y_{(t-2s)}, \cdots, \Delta Y_{(t-Ps)}$ are the seasonal differences at various seasonal lags, which are used in the seasonal autoregressive component. ε_t is the current white noise error term. $\varphi_1, \varphi_2, \cdots, \varphi_p$ are the non-seasonal autoregressive coefficients. $\Phi_1, \Phi_2, \cdots, \Phi_P$ are the seasonal autoregressive coefficients. $\theta_1, \theta_2, \cdots, \theta_q$ are the non-seasonal moving average coefficients. $\Theta_1, \Theta_2, \cdots, \Theta_Q$ are the seasonal moving average coefficients. c is a constant.

3) LSTM MODEL

An LSTM unit consists of several key components, including input gates, forget gates, output gates, and a cell state. Here's a high-level overview:

1) **Input Gate (i_t):** It determines which information from the current input should be stored in the cell state.

$$i_t = \sigma\left(W_i * \left[h_{(t-1)}, x_t\right] + b_i\right) \tag{6}$$

where,
i_t is the input gate activation at time step t.
σ represents the sigmoid activation function.
W_i is the weight matrix for the input gate.
$h_{(t-1)}$ is the previous hidden state at time step $t-1$.
x_t is the input at time step t.
'+' denotes matrix concatenation.
b_i is the bias vector for the input gate.

2) **Forget Gate (f_t):** It decides which information from the previous cell state should be forgotten or retained.

$$f_t = \sigma\left(W_f * \left[h_{(t-1)}, x_t\right] + b_f\right) \tag{7}$$

where,
f_t is the forget gate activation at time step t.
σ represents the sigmoid activation function.
W_f is the weight matrix for the forget gate.
$h_{(t-1)}$ is the previous hidden state at time step $t-1$.
x_t is the input at time step t.
'+' denotes matrix concatenation.

3) **Cell State Update (C_t):** It represents the memory of the LSTM unit and is updated using the input gate, forget gate, and a candidate cell state.

$$C_t = f_t * C_{(t-1)} + i_t * g_t \tag{8}$$

where,
C_t is the updated cell state at time step t.
f_t is the forget gate activation at time step t.
$C_{(t-1)}$ is the previous cell state at time step $t-1$.
i_t is the input gate activation at time step t.
g_t is the candidate cell state activation at time step t.

4) **Output Gate (o_t):** It controls which part of the cell state should be exposed as the output.

$$o_t = \sigma\left(W_o * \left[h_{(t-1)}, x_t\right] + b_o\right) \tag{9}$$

where,
o_t is the output gate activation at time step t.
σ represents the sigmoid activation function.
W_o is the weight matrix for the output gate.
$h_{(t-1)}$ is the previous hidden state at time step $t-1$.
x_t is the input at time step t.
'+' denotes matrix concatenation.
b_o is the bias vector for the output gate.

5) **Hidden State Update (h_t):** The hidden state 'h_t' in an LSTM is the network's internal representation of information at a specific time step 't' in a sequence.

$$h_t = o_t * tanh(C_t) \tag{10}$$

where,
h_t is the updated hidden state at time step t.
o_t is the output gate activation at time step t.
$tanh$ represents the hyperbolic tangent activation function.
C_t is the updated cell state at time step t.

The overall working principle is shown in Figure-1.

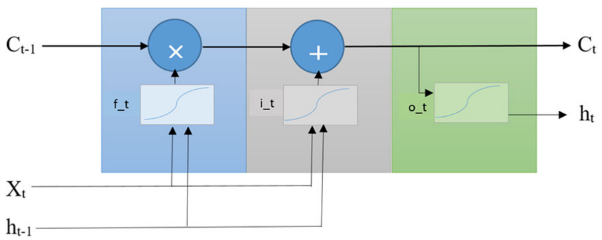

FIGURE 1. Working principle of LSTM.

B. MODEL EVALUATION METRICS

We have tried to comprehend the overall performance of the proposed model using the standard error detection formula [7], [36], [37], [38], [39] as described.

1) **Mean Absolute Percentage Error (MAPE):** MAPE, a statistical metric, gauges the accuracy of a forecasting system by expressing it as a percentage. It is computed as the average absolute percentage error for each time period, calculated as the absolute difference between forecasted and actual values divided by the actual values. The formula is outlined below.

$$M = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{A_t - F_t}{A_t}\right| \tag{11}$$

where,
N = Frequency of the summation iteration
A_t = Actual value
F_t = Forecast value

2) **Mean Squared Error (MSE):** MSE quantifies the proximity of a regression line to a given set of points. It achieves this by computing the distances from the points to the regression line and then squaring these distances. This squaring operation is necessary to work exclusively with magnitude values and also assigns greater importance to larger discrepancies. The formula for MSE is as stated below.

$$M = \frac{1}{N}\sum_{t=1}^{N}(Y_t - Y_t')2 \tag{12}$$

where,
N = Number of times the summation iteration happens
Y_t = Actual value
Y_t' = Predicted value

3) **Mean Squared Logarithmic Error (MSLE):** As implied by its name, MSLE (Mean Squared Logarithmic Error) is a modification of the Mean Squared Error. The incorporation of the logarithm in MSLE emphasizes the relative differences between actual and predicted values. The formula is outlined below.

$$L(Y, Y') = \frac{1}{N}\sum_{t=1}^{N}(log(Y_t + 1) - log(Y_t' + 1))^2 \tag{13}$$

where,
N = Number of times the summation iteration happens
Y_t = Actual value
Y_t' = Predicted value

IV. PROPOSED MODEL

In the first subsection of this research, we aimed to use the Valence Aware Dictionary for Sentiment Reasoning (VADER) [32], [34] to classify the sentiments of customer expressed in product reviews. VADER returns the level of satisfaction (S_n) of a customer about that particular product or service in a range of -1 to $+1$ (i.e. $-1 \leq S_n \leq 1$). A value of -1 represents an extremely negative satisfaction, while a value of $+1$ represents an extremely positive satisfaction. VADER provides sentiment scores within the range of -1 to 1, whereas real-world online retail platforms typically receive customer reviews (say Rn) on a scale of 1 to 5. To facilitate meaningful comparisons, a normalization process is applied to the VADER scores, transforming them into the 1 to 5 range. In this normalized score system, a score of 1 signifies an exceptionally negative level of satisfaction, while a score of 5 corresponds to an exceedingly positive degree of satisfaction. This normalization allows for a more intuitive alignment between VADER sentiment scores and the rating system used by customers, aiding in the interpretation of sentiment analysis results in practical applications. That is for each customer's rating, the proposed methodology computes the level of customer satisfaction by review text analysis and tuned the customer's rating (R_n^*) within the ranges from 0 to 5.

Thereafter, it computes the aggregated tuned customers' rating (R_{n_agg}). Further, in the second subsection, merging this aggregated tuned customers' rating (R_{n_agg}) with standard forecasting models, this proposed methodology tries to forecast sales of that particular product. Thereafter, it computes the forecasted errors using standard error detection techniques. The overall flow diagram of the proposed model is shown in Figure-2.

Algorithm for generating the sentiment scores:

Algorithm 1: Generate_Sentiment_Score

/** This algorithm takes review rating/score (R_n) in the range 1 to 5 and Review texts as an input. It performs sentiment analysis from review texts using VADER sentiment analysis tool and used it to tune customers' review rating/score. The final output of this algorithm is the normalized sentiment-enhanced review score (R_n').**/

1) **Step 1: Start**
2) **Step 2: /* Sentiment Analysis */**

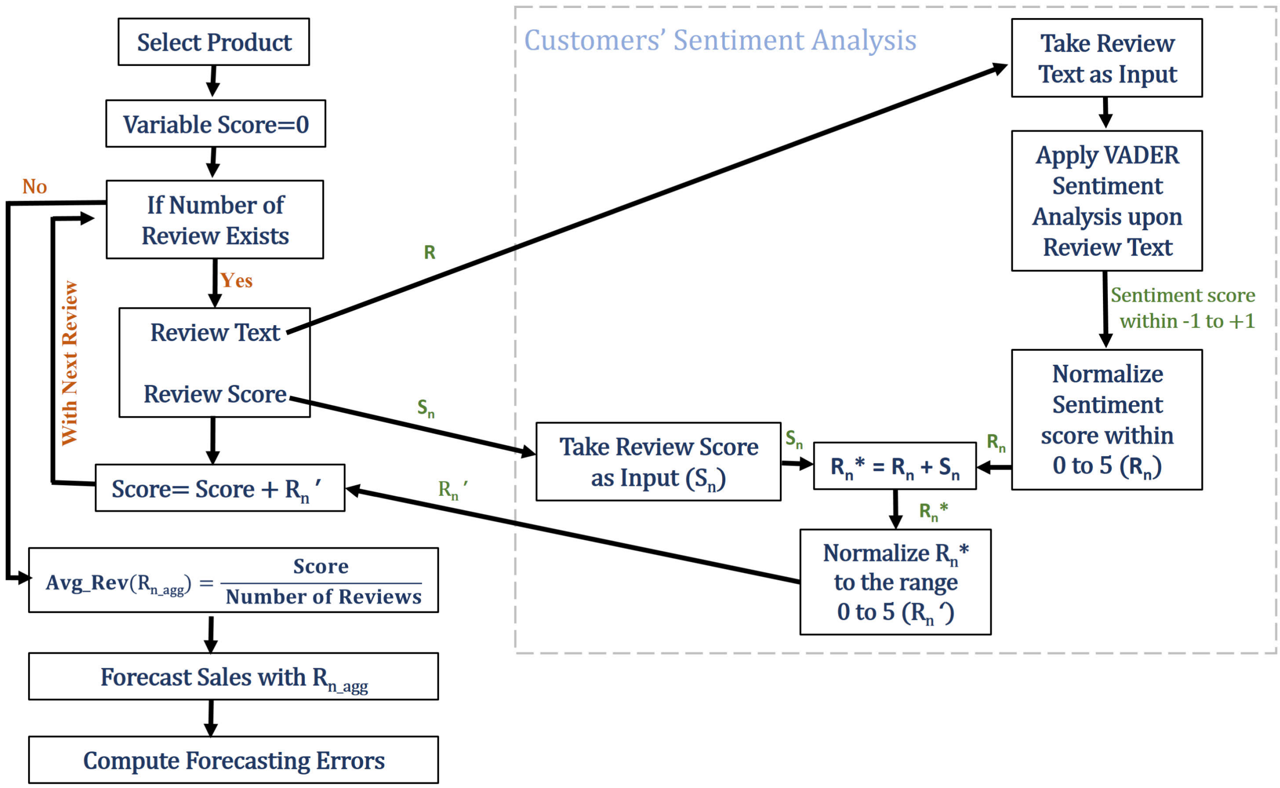

FIGURE 2. Flow diagram of the proposed model.

Perform sentiment analysis using VADER to obtain the original sentiment score (S_n).

3) **Step 3: /* Convert S_n to P_n^* */**
Map S_n to the range [0, 2] by adding 1:

$$P_n^* = S_n + 1.$$

4) **Step 4: /* Normalize P_n^* to P_n */**
Normalize P_n^* to the range [0, 5] by scaling with a factor of (5/2):

$$P_n = \frac{5}{2} * P_n^*.$$

5) **Step 5: /* Combine Sentiment Score with Review Score */**
Add the original sentiment score (S_n) to the review score (R_n) to get the new review score (R_n^*):

$$R_n^* = R_n + S_n.$$

/**The range of the term R_n^* is [0, 6]. To make it compatible with the term R_n we normalize it in **Step 6** such that $0 \leq R_n' \leq 5$.**/

6) **Step 6: /* Normalize R_n^* to R_n' */**
Normalize R_n^* to the range [0, 5] by scaling with a factor of (5/6):

$$R_n' = \frac{5}{6} * R_n^*.$$

7) **Step 7: Stop**

Algorithm for forecasting using tuned customer rating is described next.

Algorithm 2: Modified_Forecasting_of_Sales
/** This algorithm amalgamate the normalized sentiment-enhanced review score (Rn') of previous section with standard forecasting methods like ARIMA, SARIMA, LSTM to generate forecasted values. **/

1) **Step 1: Start**
/ Steps 2 to 4 involve tuning customers' review scores using the review text. **/**
2) **Step 2:** $totalReview = computeTotalNumberOfReview()$
3) **Step 3:** $averageTunedRating = 0$
4) **Step 4:** Loop for i =1 to $total\_review$
do
/* call method of subsection-1 */
$sentiment = Generate\_Sentiment\_Score()$
/** Generate_Sentiment_Score() is a method of subsection-1 **/
$averageTunedRating = averageTunedRating + sentiment$
end for

$averageTunedRating(R_{n\_agg}) = averageTunedRating \,/\, i$

5) **Step 5: /* Forecasting of sales using tuned average customers' rating */**
5.1) $Forecast_1.Values = ARIMA()$
5.2) $ForecastTuned_1.Values = ARIMA\_with\_R_{n\_agg}()$
5.3) **(a)** Compute MAPE for $Forecast_1.Values$
(b) Compute MAPE for $ForecastTuned_1.Values$
(c) Compute MSE for $Forecast_1.Values$
(d) Compute MSE for $ForecastTuned_1.Values$
(e) Compute MSLE for $Forecast_1.Values$
(f) Compute MSLE for $ForecastTuned_1.Values$
6) **Step 6:**
6.1) $Forecast_2.Values = SARIMA()$
6.2) $ForecastTuned_2.Values = SARIMA\_with\_R_{n\_agg}()$
6.3) **(a)** Compute MAPE for $Forecast_2.Values$
(b) Compute MAPE for $ForecastTuned_2.Values$
(c) Compute MSE for $Forecast_2.Values$
(d) Compute MSE for $ForecastTuned_2.Values$
(e) Compute MSLE for $Forecast_2.Values$
(f) Compute MSLE for $ForecastTuned_2.Values$
7) **Step 7:**
7.1) $Forecast_3.Values = LSTM()$
7.2) $ForecastTuned_3.Values = LSTM\_with\_R_{n\_agg}()$
7.3) **(a)** Compute MAPE for $Forecast_3.Values$
(b) Compute MAPE for $ForecastTuned_3.Values$
(c) Compute MSE for $Forecast_3.Values$
(d) Compute MSE for $ForecastTuned_3.Values$
(e) Compute MSLE for $Forecast_3.Values$
(f) Compute MSLE for $ForecastTuned_3.Values$
8) **Step 8: Stop**

V. CASE STUDY RESULTS

We have implemented our proposed model on the review data of Amazon [35], a renowned online retailer. The experimental data spanning over a period of 22 years. Data available in "https://nijianmo.github.io/amazon/index.html". Since the data set consists solely of historical records, specifically from May 1996 to October 2018 (the time period may differ for each data set), we have divided it into two subsets: training data (the first 80% of the dataset in chronological order) and test data (the remaining 20%).

In this case study, we tries to compute forecasted errors for all type (small, medium and large) of data sets and hence have selected data sets accordingly. Further, we have reported our results with 4 different cases. Such as (i) Considering without any review scores, (ii) Considering only sentiment score (S_n), (iii) Considering only review score (Rn), and (iv) Considering the proposed combination of both the sentiment score (Sn) and review score (Rn) that is R_n'.

The experiments were conducted on a Windows machine having Intel Core-i5 processor, 16 GB of RAM, and a 250 GB of SSD. The experimental environment used the following software packages: Jupyter Notebook and Python 3.11.3 for coding, pandas for manipulating large data, matplotlib and seaborn for visualizing data, tensorflow and sklearn are used for machine learning and statistical modelling, nltk and SentimentIntensityAnalyzer used for natural language processing.

A. CASE STUDY-1 (NUMBER OF REVIEWS 89656)

We have started with a relatively small data set "Magazine_Subscriptions" as our first case study. Here the average customers' rating is 3.991. But, according to the proposed model, when this average customers' rating is tuned with customers' review text, it becomes 3.42. Blending this tuned average customers' rating with ARIMA, SARIMA and LSTM we have tried to forecast sales of that particular product using the proposed model.

Figure-3 presents an illustrative graphical representation elucidating the relationship between "Time" and "Sales." Additionally, a forecasted trend generated through the ARIMA model enhances insights into future sales patterns over time.

Thereafter, we compute the forecasting errors using standard error detection techniques such as MAPE, MSE and MSLE [7], [36], [37]. The computed error values are shown in Table-1 which shows that in all the cases, the proposed model (ARIMA-with tuned customers' rating) producing less forecasting errors.

TABLE 1. ARIMA forecasted errors (Case study-1).

Error Detection Technique	ARIMA	ARIMA-with customers' sentiment score (S_n)	ARIMA-with customers' review score (R_n)	ARIMA-with tuned customers' rating (R_n') (Proposed Model)
MAPE	248.055	183.084	187.639	**157.437**
MSE	478.668	286.244	294.028	**262.270**
MSLE	0.855	0.638	0.651	**0.568**

TABLE 2. SARIMA forecasted errors (Case study-1).

Error Detection Technique	SARIMA	SARIMA-with customers' sentiment score (S_n)	SARIMA-with customers' review score (R_n)	SARIMA-with tuned customers' rating (R_n') (Proposed Model)
MAPE	249.440	184.041	188.628	**158.187**
MSE	484.929	288.067	296.124	**262.657**
MSLE	5.256	0.641	0.654	**0.570**

Therefore in this case, the forecasted errors are minimum using proposed model in all the cases as shown in Table-1.

Subsequently, the identical procedure is repeated for SARIMA. Figure-4 exhibits the graphical representation of "Time" versus "Sales," accompanied by a forecasted graph generated through SARIMA analysis.

Thereafter, in this case also, we compute the forecasting errors using MAPE, MSE and MSLE. The computed

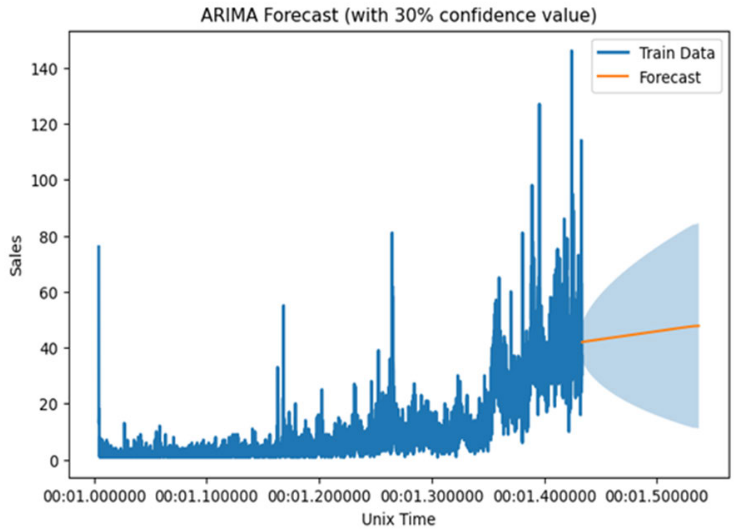

FIGURE 3. Forecasting graph using ARIMA (Case study-1).

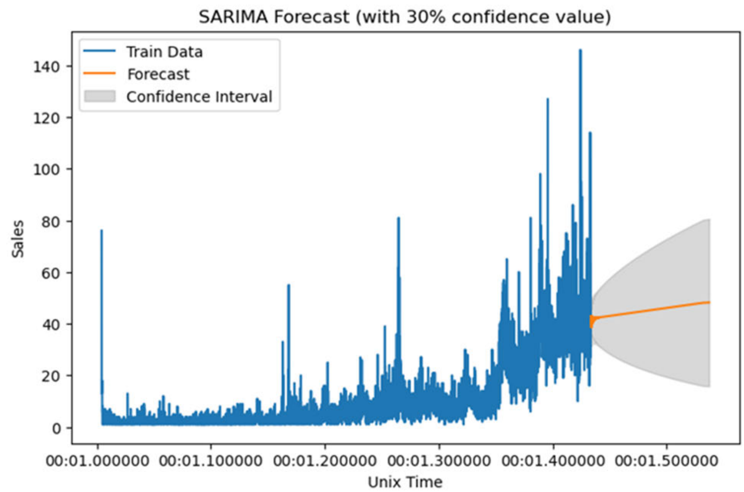

FIGURE 4. Forecasting graph using SARIMA (Case study-1).

error values are shown in Table-2 which shows that the proposed SARIMA-with tuned customers' rating producing less forecasting error in all the cases.

Continuing, the process extends to LSTM. Figure-5 showcases the relationship between "Time" and "Sales," alongside a forecasted graph produced through LSTM, providing further insights into sales forecasts over time.

Thereafter, in this case also, we compute the forecasting errors using MAPE, MSE and MSLE. The computed error values are shown in Table-3 which shows that the proposed LSTM-with tuned customers' rating producing less forecasting error in all the cases.

B. CASE STUDY-2 (NUMBER OF REVIEWS 882403)

Next, we have selected a medium data set "AMAZON_FASHION" as our next case study. Here the average customers' rating is 3.888. But, according to the proposed model, when this average customers' rating is tuned with customers' review text, it becomes 3.323. Blending this tuned average customers' rating with ARIMA, SARIMA and LSTM we have tried to forecast sales of that particular product using the proposed model.

TABLE 3. LSTM forecasted errors (Case study-1).

Error Detection Technique	LSTM	LSTM-with customers' sentiment score (S_n)	LSTM-with customers' review score (R_n)	LSTM-with tuned customers' rating (R'_n) (Proposed Model)
MAPE	166.038	132.282	134.537	**119.910**
MSE	387.686	288.067	296.124	**262.657**
MSLE	0.894	0.641	0.655	**0.570**

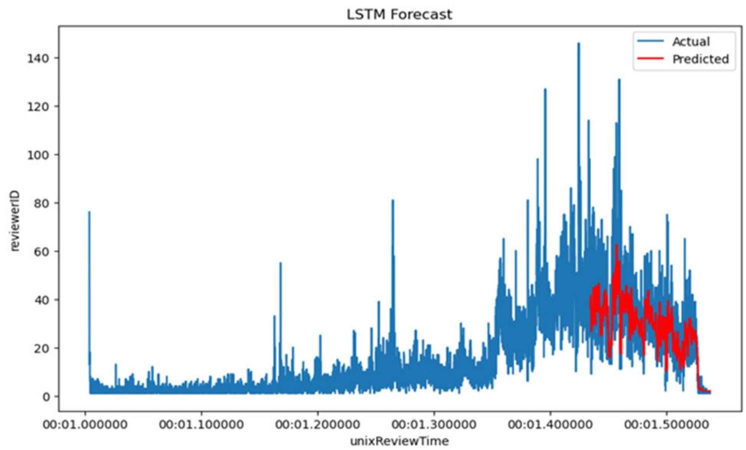

FIGURE 5. Forecasting graph using LSTM (Case study-1).

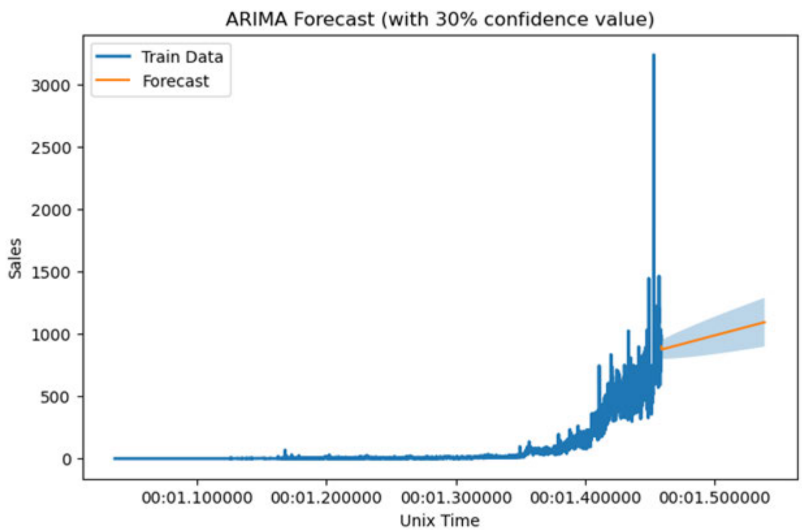

FIGURE 6. Forecasting graph using ARIMA (Case study-2).

Figure-6 depicts a detailed graphical representation illustrating the correlation between "Time" and "Sales." Furthermore, it showcases the forecasted trend derived from the ARIMA model, offering valuable insights into future sales patterns over time.

Thereafter, we compute the forecasting errors using standard error detection techniques such as MAPE, MSE and MSLE [7], [36], [37]. The computed error values are shown in Table-4 which shows that in all the cases, the proposed model (ARIMA-with tuned customers' rating) producing less forecasting errors.

Therefore in this case, the forecasted errors are minimum using proposed model in all the cases as shown in Table-4.

We apply the same methodology to SARIMA. Figure-7 provides a comprehensive view of the relationship between "Time" and "Sales," presenting both historical data and forecasted trends using SARIMA. This graphical representation offers valuable insights into sales patterns over time, enhancing our understanding of future trajectories.

TABLE 4. ARIMA forecasted errors (Case study-2).

Error Detection Technique	ARIMA	ARIMA-with customers' sentiment score (S_n)	ARIMA-with customers' review score (R_n)	ARIMA-with tuned customers' rating (R'_n) (Proposed Model)
MAPE	271.582	186.880	195.777	**160.546**
MSE	333639.921	160528.880	173642.775	**129397.147**
MSLE	1.353	0.930	0.974	**0.802**

Thereafter, in this case also, we compute the forecasting errors using MAPE, MSE and MSLE. The computed error values are shown in Table-5 which shows that the proposed SARIMA-with tuned customers' rating producing less forecasting error in all the cases.

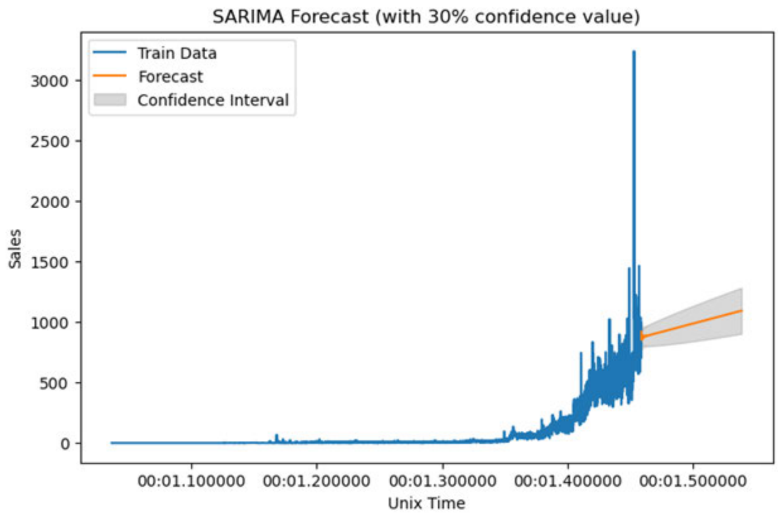

FIGURE 7. Forecasting graph using SARIMA (Case study-2).

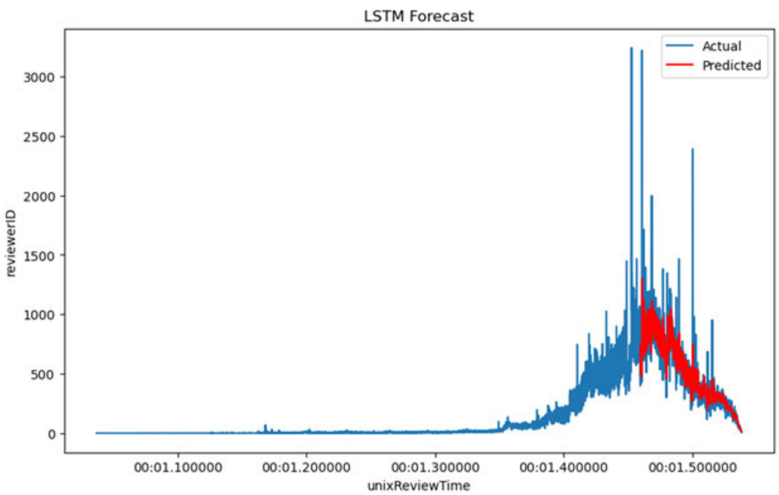

FIGURE 8. Forecasting graph using LSTM (Case study-2).

TABLE 5. SARIMA forecasted errors (Case study-2).

Error Detection Technique	SARIMA	SARIMA-with customers' sentiment score (S_n)	SARIMA-with customers' review score (R_n)	SARIMA-with tuned customers' rating (R'_n) (Proposed Model)
MAPE	249.440	184.041	188.628	**158.187**
MSE	484.929	288.067	296.124	**262.657**
MSLE	5.256	0.641	0.654	**0.570**

Continuing onward, the process expands to LSTM. Figure-8 exhibits the correlation between "Time" and "Sales," presenting a forecasted graph derived from LSTM, offering deeper insights into future sales projections over time.

Thereafter, in this case also, we compute the forecasting errors using MAPE, MSE and MSLE. The computed error values are shown in Table-6 which shows that the proposed LSTM-with tuned customers' rating producing less forecasting error in all the cases.

TABLE 6. LSTM forecasted errors (Case study-2).

Error Detection Technique	LSTM	LSTM-with customers' sentiment score (S_n)	LSTM-with customers' review score (R_n)	LSTM-with tuned customers' rating (R'_n) (Proposed Model)
MAPE	135.812	104.517	107.587	**95.754**
MSE	180939.624	160420.681	173516.359	**129338.851**
MSLE	1.073	0.930	0.974	**0.802**

C. CASE STUDY-3 (NUMBER OF REVIEWS 10,053,882)

Next, we have selected a very large data set "CELLPHONES AND ACCESSORIES" from the same data source as our

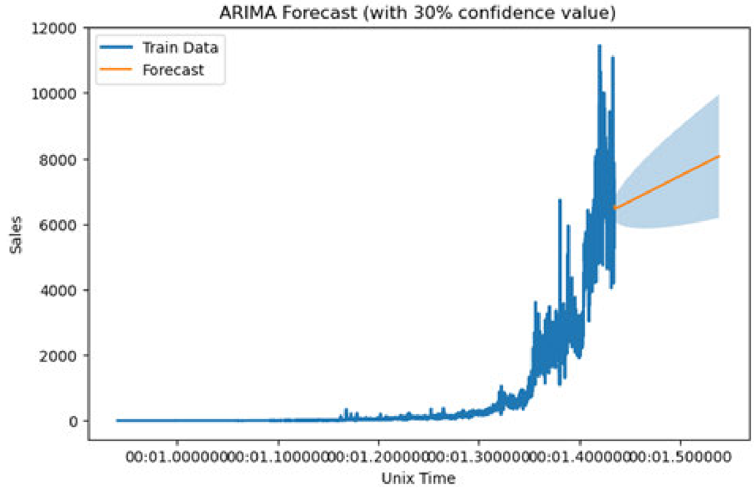

FIGURE 9. Forecasting graph using ARIMA (Case study-3).

next case study. Here the average customers' rating is 3.915. But, according to the proposed model, when this average customers' rating is tuned with customers' review text, it becomes 3.339. Blending this tuned average customers' rating with ARIMA, SARIMA and LSTM we have tried to forecast sales of that particular product using the proposed model.

Figure-9 illustrates the correlation between "Time" and "Sales" graphically. Additionally, it presents the forecasted trend using the ARIMA model, providing insights into future sales patterns.

Thereafter, in the same way, we compute the forecasting errors. The computed error values are shown in Table-7. According to the Table-7,in this case also, the proposed model (ARIMA-with tuned customers' rating) producing less forecasting errors in all the cases.

TABLE 7. ARIMA forecasted errors (Case study-3).

Error Detection Technique	ARIMA	ARIMA-with customers' sentiment score (S_n)	ARIMA-with customers' review score (R_n)	ARIMA-with tuned customers' rating (R'_n) (Proposed Model)
MAPE	271.582	186.880	195.777	**160.546**
MSE	333639.921	160528.880	173642.775	**129397.147**
MSLE	1.353	0.930	0.974	**0.802**

Therefore in this case, the forecasted errors are minimum using proposed model in all the cases as shown in Table–7.

We use the same approach by SARIMA. Figure-10 provides a comprehensive depiction of the relationship between "Time" and "Sales," showcasing both historical data and forecasted trends generated by SARIMA. This visualization enhances our understanding of sales patterns, facilitating anticipation of future trajectories.

Thereafter, in this case also, we compute the forecasting errors using MAPE, MSE and MSLE. The computed error values are shown in Table-8 which shows that the proposed SARIMA-with tuned customers' rating producing less forecasting error in all the cases.

TABLE 8. SARIMA forecasted errors (Case study-3).

Error Detection Technique	SARIMA	SARIMA-with customers' sentiment score (S_n)	SARIMA-with customers' review score (R_n)	SARIMA-with tuned customers' rating (R'_n) (Proposed Model)
MAPE	934.209	674.122	723.443	**615.944**
MSE	11809204.845	6283916.066	6754053.874	**6105110.817**
MSLE	15.063	0.591	0.631	**0.553**

We apply the same methodology by LSTM. In Figure-11, a comprehensive illustration of the correlation between "Time" and "Sales" is presented, featuring both historical data and LSTM-generated forecasted trends. This visualization enhances our understanding of sales patterns, aiding in forecasting future trajectories.

Thereafter, in this case also, we compute the forecasting errors using MAPE, MSE and MSLE. The computed error values are shown in Table-9 which shows that the proposed LSTM-with tuned customers' rating producing less forecasting error in all the cases.

VI. DISCUSSION

In this section, the results from the case studies are analyzed to assess the outcomes of this research. The aim is to gain a better understanding of the improvements over existing methodologies and identify instances where the proposed methodology may not be applicable.

In the case of a small dataset (Case Study-1), where the average customers' rating is 3.991 and the tuned average customers' rating is 3.42, the proposed methodology producing a significant improvement. Figure-12 displays a

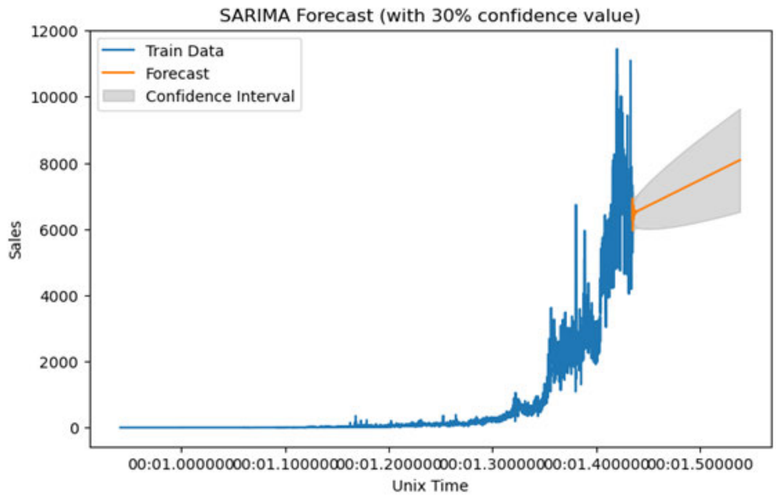

FIGURE 10. Forecasting graph using SARIMA (Case study-3).

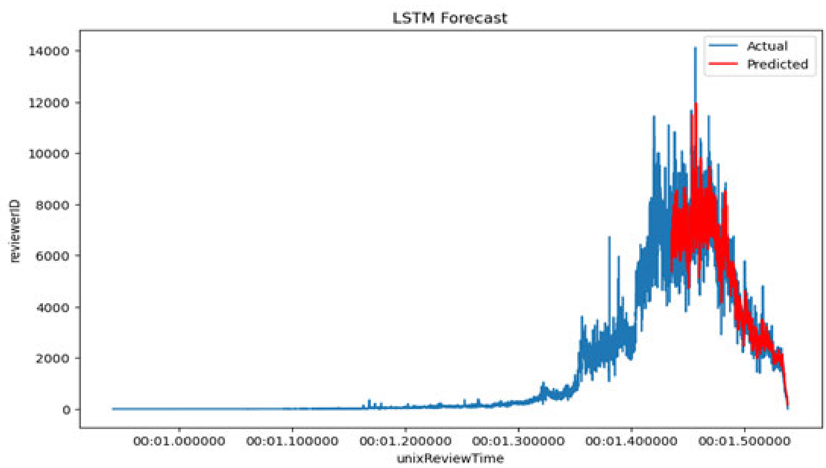

FIGURE 11. Forecasting graph using LSTM (Case study-3).

TABLE 9. LSTM forecasted errors (Case study-3).

Error Detection Technique	LSTM	LSTM-with customers' sentiment score (S_n)	LSTM-with customers' review score (R_n)	LSTM-with tuned customers' rating (R'_n) (Proposed Model)
MAPE	647.431	477.393	509.612	**439.436**
MSE	11429262.803	6283916.066	6754053.874	**6105110.817**
MSLE	0. 897	0. 591	0. 631	**0. 553**

statistical comparison of forecasting errors based on the reference Table-1,2,3 in Section-V.

Therefore, in the case of small sized overrated products, depending upon various forecasting methods and customers' rating tuning technique, we are getting around 10% to 58% improvement when the customers' rating are tuned according to the proposed method.

We perform the same result-set analysis in the case of a medium sized dataset (Case Study-2). In this instance, the average rating from customers is 3.888, while the tuned average customers' rating stands at 3.323. In this scenario, the proposed methodology demonstrates a substantial improvement, as illustrated by the statistical comparison in the accompanying Figure-13.

Hence from the Figure-13 we can conclude that, for a medium-sized products that are overrated, depending upon various forecasting methods, the proposed technique results in an enhancement ranging from 25% to 93%.

In case of large sized dataset (Case Study-3), where the average rating from customers is 3.915 and tuned average customers' rating is 3.339, the statistical comparison as shown in Figure-14 clearly depicts the advantage of the proposed method.

Therefore from the Figure-14, we can conclude that for a large-sized product that are overrated, depending

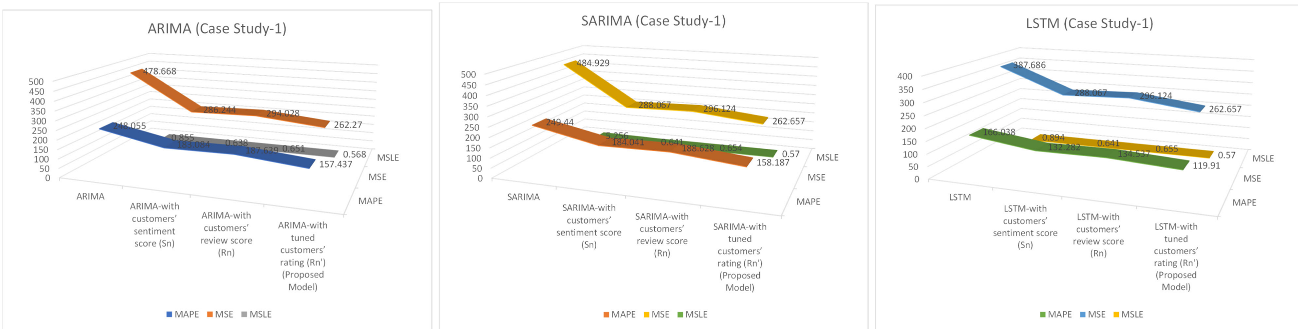

FIGURE 12. Comparison of different 3 methods ARIMA, SARIMA, LSTM (Case study-1).

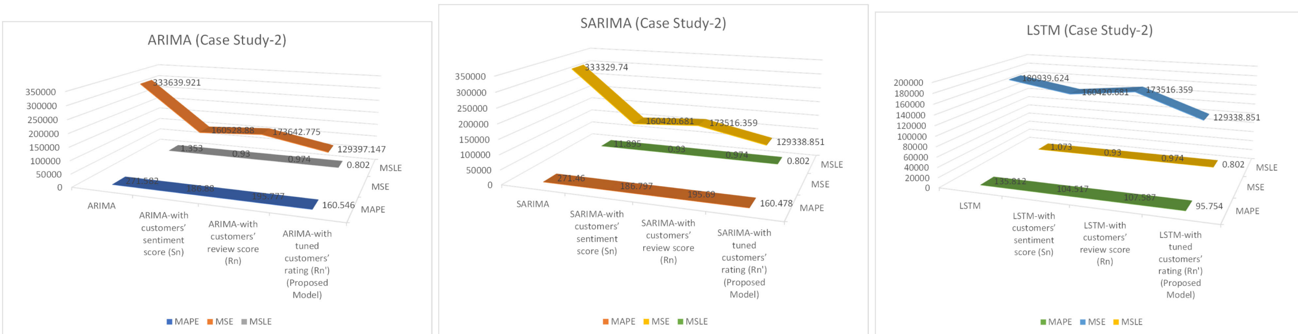

FIGURE 13. Comparison of different 3 methods ARIMA, SARIMA, LSTM (Case study-2).

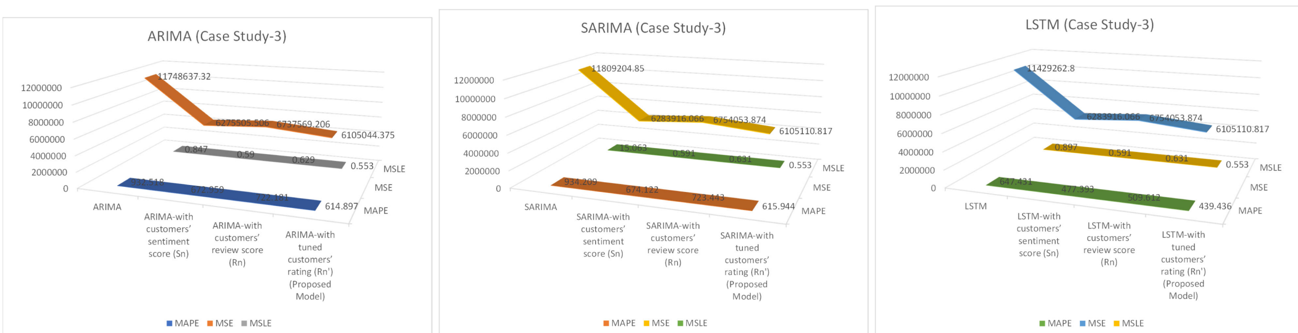

FIGURE 14. Comparison of different 3 methods ARIMA, SARIMA, LSTM (Case study-3).

upon various forecasting methods, the proposed technique results in an enhancement ranging from 32% to 96%.

Hence, form the Figure 12, 13 and 14, it is evident that the proposed methodology yields better result for any types (small, medium, large) of overrated product. But, when the product is not overrated by the customers', for example the dataset "Appliances" or "Industrial_and_Scientific", the proposed methodology is not producing always a better result. Therefore, we can conclude that for any types of overrated product, the proposed methodology produces better result for any types of standard forecasting method.

VII. CONCLUSION AND FUTURE SCOPE

In this research work, overrating of product is identified as a problem that misleads about the acceptance of a product, which also affects the sales forecasting model and may even lead to business loss. This problem is analyzed in details, and the solution is proposed by amalgamating both the review score and review text of a product for precise sales forecasting. The limitation of giving ratings in integer form is managed here by incorporating the textual review, which reflects the customer's actual view about the product. The sentiment score derived from the textual analysis is seamlessly integrated with the numerical review scores provided by customers. This amalgamation process synthesizes quantitative and qualitative insights, culminating the generation of a tuned review score for the product. Next, it uses that tuned review score of the product for forecasting sales of that particular product with the help of standard forecasting methods. Experiments are conducted on 3 different dataset of small, medium, and large volumes to check the efficacy of the proposed model. Depending on the datasets, the proposed model offers enhancements ranging from 10% to 96% over existing forecasting methods. Therefore, this model proves effective for diverse datasets with varying volume and variety.

Further enhancement of the work will involve exploring different NLP and LLM models for text processing based on various types of datasets. Additionally, experimentation with new forecasting models that are evolving may lead to improvements in results.

REFERENCES

[1] R. Kalla, S. Murikinjeri, and R. Abbaiah, "An improved demand forecasting with limited historical sales data," in *Proc. Int. Conf. Comput. Commun. Informat. (ICCCI)*, Jan. 2020, pp. 1–5.

[2] S. U. Tareq, M. H. Noor, and C. Bepery, "Framework of dynamic recommendation system for e-shopping," *Int. J. Inf. Technol.*, vol. 12, no. 1, pp. 135–140, Mar. 2020.

[3] S. Singh, S. Hussain, and M. A. Bazaz, "Short term load forecasting using artificial neural network," in *Proc. 4th Int. Conf. Image Inf. Process. (ICIIP)*, Dec. 2017, pp. 1–5.

[4] P. Kroha, R. Baeza-Yates, and B. Krellner, "Text mining of business News for forecasting," in *Proc. 17th Int. Conf. Database Expert Syst. Appl. (DEXA06)*, 2006, pp. 171–175.

[5] M. A. Khan, S. Saqib, T. Alyas, A. Ur Rehman, Y. Saeed, A. Zeb, M. Zareei, and E. M. Mohamed, "Effective demand forecasting model using business intelligence empowered with machine learning," *IEEE Access*, vol. 8, pp. 116013–116023, 2020.

[6] G. H. F. M. Oliveira, R. C. Cavalcante, G. G. Cabral, L. L. Minku, and A. L. I. Oliveira, "Time series forecasting in the presence of concept drift: A PSO-based approach," in *Proc. IEEE 29th Int. Conf. Tools Artif. Intell. (ICTAI)*, Nov. 2017, pp. 239–246.

[7] U. M. Sirisha, M. C. Belavagi, and G. Attigeri, "Profit prediction using ARIMA, SARIMA and LSTM models in time series forecasting: A comparison," *IEEE Access*, vol. 10, pp. 124715–124727, 2022.

[8] P. Ghosh, L. J. Ghosh, N. C. Debnath, and S. Sen, "Reducing bullwhip effect in distributed supply chain management by virtual data warehouse and modified-prophet," in *Proc. Int. Conf. Comput. Intell. Comput.* Cham, Switzerland: Springer, 2020, pp. 223–232.

[9] P. Ghosh, D. Sadhu, J. K. Mandal, N. C. Debnath, and S. Sen, "RHProphet: An enhanced sales forecasting model," *Int. J. Comput. Appl.*, vol. 28, no. 4, pp. 1–10, 2021.

[10] P. Ghosh, D. Sadhu, and S. Sen, "A real-time business analysis framework using virtual data warehouse," *Int. Arab J. Inf. Technol.*, vol. 18, no. 4, pp. 585–595, 2021.

[11] B. Singh, P. Kumar, N. Sharma, and K. P. Sharma, "Sales forecast for Amazon sales with time series modeling," in *Proc. 1st Int. Conf. Power, Control Comput. Technol. (ICPC2T)*, Jan. 2020, pp. 38–43.

[12] Q. Chen, Y. Wu, and X. Chen, "Research on customers demand forecasting for E-business web site based on LS-SVM," in *Proc. Int. Symp. Electron. Commerce Secur.*, 2008, pp. 66–70.

[13] L. Dong, Z. Hua, L. Huang, T. Ji, F. Jiang, G. Tan, and J. Zhang, "The impacts of live chat on service–product purchase: Evidence from a large online outsourcing platform," *Inf. Manage.*, vol. 61, no. 3, Apr. 2024, Art. no. 103931.

[14] J. Deveikyte, H. Geman, C. Piccari, and A. Provetti, "A sentiment analysis approach to the prediction of market volatility," *Frontiers Artif. Intell.*, vol. 5, Dec. 2022, Art. no. 836809.

[15] S. J. Taylor and B. Letham, "Forecasting at scale," *Amer. Statistician*, vol. 72, no. 1, pp. 37–45, 2018.

[16] S. L. Ho and M. Xie, "The use of ARIMA models for reliability forecasting and analysis," *Comput. Ind. Eng.*, vol. 35, nos. 1–2, pp. 213–216, Oct. 1998.

[17] J. Fattah, L. Ezzine, Z. Aman, H. E. Moussami, and A. Lachhab, "Forecasting of demand using ARIMA model," *Int. J. Eng. Bus. Manage.*, vol. 10, Jun. 2018, Art. no. 1847979018808673.

[18] S. I. Vagropoulos, G. I. Chouliaras, E. G. Kardakos, C. K. Simoglou, and A. G. Bakirtzis, "Comparison of SARIMAX, SARIMA, modified SARIMA and ANN-based models for short-term PV generation forecasting," in *Proc. IEEE Int. Energy Conf. (ENERGYCON)*, Apr. 2016, pp. 1–6.

[19] M. Olsson and L. Soder, "Modeling real-time balancing power market prices using combined SARIMA and Markov processes," *IEEE Trans. Power Syst.*, vol. 23, no. 2, pp. 443–450, May 2008.

[20] Q. Chen, W. Zhang, and Y. Lou, "Forecasting stock prices using a hybrid deep learning model integrating attention mechanism, multi-layer perceptron, and bidirectional long-short term memory neural network," *IEEE Access*, vol. 8, pp. 117365–117376, 2020.

[21] T. Sanboon, K. Keatruangkamala, and S. Jaiyen, "A deep learning model for predicting buy and sell recommendations in stock exchange of Thailand using long short-term memory," in *Proc. IEEE 4th Int. Conf. Comput. Commun. Syst. (ICCCS)*, Feb. 2019, pp. 757–760.

[22] W.-X. Fang, P.-C. Lan, W.-R. Lin, H.-C. Chang, H.-Y. Chang, and Y.-H. Wang, "Combine Facebook prophet and LSTM with BPNN forecasting financial markets: The Morgan Taiwan index," in *Proc. Int. Symp. Intell. Signal Process. Commun. Syst. (ISPACS)*, Dec. 2019, pp. 1–2.

[23] Y. Xu, E. Wang, Y. Yang, and Y. Chang, "A unified collaborative representation learning for neural-network based recommender systems," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 11, pp. 5126–5139, Nov. 2022.

[24] B. He and L. Yin, "Prediction modelling of cold chain logistics demand based on data mining algorithm," *Math. Problems Eng.*, vol. 2021, pp. 1–9, Feb. 2021.

[25] G. Papacharalampous, H. Tyralis, and D. Koutsoyiannis, "Predictability of monthly temperature and precipitation using automatic time series forecasting methods," *Acta Geophysica*, vol. 66, no. 4, pp. 807–831, Aug. 2018.

[26] B. Liu, M. Li, Z. Ji, H. Li, and J. Luo, "Intelligent productivity transformation: Corporate market demand forecasting with the aid of an AI virtual assistant," *J. Organizational End User Comput.*, vol. 36, no. 1, pp. 1–27, Jan. 2024.

[27] M. Mangla, N. Sharma, and S. N. Mohanty, "A sequential ensemble model for software fault prediction," *Innov. Syst. Softw. Eng.*, vol. 18, no. 2, pp. 301–308, Jun. 2022.

[28] H. Iftikhar, J. E. Turpo-Chaparro, P. C. Rodrigues, and J. L. López-Gonzales, "Day-ahead electricity demand forecasting using a novel decomposition combination method," *Energies*, vol. 16, no. 18, p. 6675, Sep. 2023.

[29] I. Shah, H. Iftikhar, S. Ali, and D. Wang, "Short-term electricity demand forecasting using ComponentsEstimation technique," *Energies*, vol. 12, no. 13, p. 2532, Jul. 2019.

[30] P. Ghosh, S. Som, and S. Sen, "Business intelligence development by analysing customer sentiment," in *Proc. 7th Int. Conf. Rel., INFOCOM Technol. Optim., Trends Future Direction (ICRITO)*, Aug. 2018, pp. 287–290.

[31] J. Ni, J. Li, and J. McAuley, "Justifying recommendations using distantly-labeled reviews and fine-grained aspects," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 188–197.

[32] C. Hutto and E. Gilbert, "VADER: A parsimonious rule-based model for sentiment analysis of social media text," in *Proc. Int. AAAI Conf. Web Social Media*, vol. 8, May 2014, pp. 216–225.

[33] A. P. Ekaputri and S. Akbar, "Financial news sentiment analysis using modified VADER for stock price prediction," in *Proc. 9th Int. Conf. Adv. Inform., Concepts, Theory Appl. (ICAICTA)*, Sep. 2022, pp. 1–6.

[34] K. P. Nitha and S. Sivakumari, "Sentimental analysis for sensex stocks using VADER algorithm—An approach for investment recommendations in Indian stock market," *Turkish Online J. Qualitative Inquiry*, vol. 12, no. 6, pp. 1700–1707, 2021.

[35] J. Ni. (2018). *Amazon Review Data (2018)*. Accessed: Jan. 3, 2024. [Online]. Available: https://nijianmo.github.io/amazon/index.html/

[36] "Virtual cellular manufacturing," in *Encyclopedia of Production and Manufacturing Management*, P. M. Swamidass, Ed. Boston, MA, USA: Springer, 2000, doi: 10.1007/1-4020-0612-8_1037.

[37] J. Tayman and D. A. Swanson, "On the validity of MAPE as a measure of population forecast accuracy," *Population Res. Policy Rev.*, vol. 18, pp. 299–322, May 1999.

[38] A. Nair and S. K. Joshi, "Short term load forecasting using probabilistic neural network based algorithm," in *Proc. Int. Conf. Comput. Intell. Commun. Netw.*, Nov. 2010, pp. 128–132.

[39] Q. Li and M. Yu, "Achieving sales forecasting with higher accuracy and efficiency: A new model based on modified transformer," *J. Theor. Appl. Electron. Commerce Res.*, vol. 18, no. 4, pp. 1990–2006, Nov. 2023.

PARTHA GHOSH received the M.Tech. and Ph.D. (Tech.) degrees in information technology from the University of Calcutta, Kolkata, India, in 2013 and 2023, respectively. He is currently an Assistant Professor with the Department of Computer Science and Engineering, Academy of Technology, under the Maulana Abul Kalam Azad University of Technology, West Bengal, India. He has published 26 research articles in various reputed peer-reviewed journals, international conferences, and book chapters. His research interests include optimization techniques, data warehousing, big data analysis, machine learning, and business intelligence. He has been a member of the technical program committees for several international journal and conferences. He is a Lifetime Member of IETE. He has served as a reviewer for numerous international journals and conferences.

OENDRILA SAMANTA received the Master of Computer Application degree from West Bengal University of Technology, in 2008, and the Ph.D. degree in Computer Science and Engineering from the Indian Statistical Institute, Kolkata, in 2019. She is currently an Assistant Professor with the Department of Computer Science and Engineering, Academy of Technology, West Bengal, under the Maulana Abul Kalam Azad University of Technology. Her research interests include pattern recognition, natural language processing, and machine learning.

TAKAAKI GOTO (Member, IEEE) received the Doctor of Engineering degree from Toyo University, Japan, in 2009. He was a Project Assistant Professor with The University of Electro-Communications, Japan, from 2009 to 2015. He was an Associate Professor with Ryutsu Keizai University, Japan, from 2015 to 2019. He is currently an Associate Professor with Toyo University. His main research interests include applications of graph grammars, visual languages, and software development environments. He is a member of ACM, ACIS, ISCA, IEICE, and IPSJ.

SOUMYA SEN (Member, IEEE) is currently an Assistant Professor with the A. K. Choudhury School of Information Technology, University of Calcutta, India. He has been granted three international patents in the USA, Japan, and South Korea, and also two Indian copyrights. He has authored two books and also successfully supervised two Ph.D. students. He has published more than 120 research papers in international peer-reviewed journals and conferences. His research interests include data warehouses, OLAP tools, machine learning, business intelligence, and distributed databases. He was a technical program committee member of several international conferences across the world. He also acted as a reviewer of many international journals.