Journal of Theoretical and Applied Information Technology
15th February 2026. Vol.104. No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

QUANTIFYING THE UNCERTAINTY OF FINANCIAL DISTRESS SEARCH BEHAVIOR: A HYBRID LSTM AND MONTE CARLO DROPOUT APPROACH FOR THE PAWNING INDUSTRY

WANDA WANDOKO[1] , ROBERTUS NUGROHO PERWIRO ATMOJO[2], IGNATIUS ENDA PANGGATI[3]

[12]Bina Nusantara University, Information System Department, Jakarta, Indonesia

[3]Bina Nusantara University, Management Department, Jakarta, Indonesia

E-mail: [1]wwandoko@binus.ac.id, [2]ratmojo@binus.ac.id, [3]enda@binus.ac.id

ABSTRACT

This study aims to develop a decision support model to predict public interest trends in Indonesian pawning services while simultaneously quantifying uncertainty risks through a hybrid framework. Utilizing daily Google Trends time-series data as a real-time proxy for pawning demand over a five-year period (November 2020–October 2025), this research develops a hybrid model. It integrates Long Short-Term Memory (LSTM) to map complex non-linear patterns and long-term dependencies, with Monte Carlo simulation (via the MC Dropout technique) to generate probabilistic forecasting. Empirical results demonstrate that the deterministic LSTM component achieved "Highly Accurate" performance, measured by a Mean Absolute Percentage Error (MAPE) of 7.90% on the test set. The MAPE results prove the model's capability in separating fundamental trend signals from daily noise. Furthermore, the probabilistic Monte Carlo component successfully transformed single-point forecasts into measurable risk distributions. The model proved effective as an anomaly detector, identifying market surprises when actual values fell outside the 95% Confidence Interval. The novelty of this research lies in the hybrid integration of deep learning and stochastic Monte Carlo simulation applied to the underexplored pawning financial sector. This study contributes theoretically to time-series forecasting and pawning literature. Additionally, it offers managerial contributions, particularly for stakeholders in the pawning industry, by providing a tool for risk-based decision-making.

Keywords: *Time-Series Forecasting, Long Short-Term Memory (LSTM), Monte Carlo Simulation, Google Trends, Pawning Industry, Risk Management.*

1. INTRODUCTION

Currently The global financial services industry is undergoing a fundamental transformation driven by digitalization. Amidst this evolution, access to rapid and secure short-term liquidity, such as pawning services, remains a fundamental need for household financial stability and Micro, Small, and Medium Enterprises (MSMEs). As one of Southeast Asia's largest financial markets, Indonesia's "pawning" (gadai) industry plays a crucial, yet often overlooked, role. The pawnbroking industry in Indonesia is showing positive growth in 2025, with the number of companies growing by 12%, total assets increasing by 23%, and revenue growing significantly by 37.5% [1].

Financial markets are frequently confronted with information asymmetry, wherein consumers' financial intentions cannot be directly observed by service providers. Within this context, digital information search activity is not merely construed as technical internet behavior, but rather as a 'signal' emitted by consumers acting as economic agents [2]. When consumers experience liquidity stress, prior to executing a physical transaction, such as visiting a pawnshop, they first emit digital signals via search engines [3]. Consequently, a surge in the search volume for the keyword 'gadai' in Google Trend can be theoretically validated as a leading indicator of financial distress [4]. The methodological challenge, however, is that this digital signal is often conflated with information noise, necessitating a deep learning approach to isolate the pure pawning intention signal from random search volatility [5], [6].

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The fluctuation in Google Trend is not random; it reflects complex patterns of seasonal demand (e.g., New School Year, Eid al-Fitr) and sudden economic shocks (e.g., crises, recessions, or layoffs). This high volatility creates significant operational and financial risks for industry stakeholders. The inability to accurately predict demand surges or declines hinders liquidity planning, resource allocation, and marketing strategies. To address this volatility, traditional forecasting often relies on linear statistical models such as ARIMA or SARIMA [7]–[9]. However, these traditional models fail to capture the complex non-linear patterns driven by dynamic events. Consequently, Deep Learning techniques such as Gated Recurrent Units (GRU) [10], [11] and Convolutional Neural Networks (CNN) [12], [13] have also been adopted to capture non-linear patterns that traditional statistics miss. Unlike CNNs or GRUs, LSTM possesses a superior capacity to preserve long-term dependencies without information degradation, rendering it more precise in capturing the complex volatility patterns of financial data. However, these deep learning models share a fundamental limitation: they are deterministic. They provide only a single point forecast

In reality, financial managers, particularly in the pawning industry, require more than just high-accuracy point predictions; they critically need measurable risk scenarios. These probabilistic scenarios are essential for effectively allocating liquidity reserves and formulating strategic defenses against sudden demand surges. This specific gap is addressed in this study through the hybridization of LSTM with Monte Carlo (MC) Dropout.

This creates a critical gap for financial managers who require not just a prediction of value, but a measurable scenario of risk to allocate liquidity reserves effectively. Consequently, the primary objective of this research is to address this specific gap by developing a hybrid framework that integrates the pattern-recognition power of LSTM with the uncertainty quantification of Monte Carlo (MC) Dropout. By doing so, this study aims to transform the forecasting output from a static single value into a dynamic probabilistic distribution, providing a robust decision-support tool for the Indonesian pawning industry.

2. LITERATURE REVIEW

2.1. Long Short Term Memory (LSTM)

Long Short-Term Memory (LSTM) is an advanced Recurrent Neural Network (RNN) architecture introduced by Hochreiter & Schmidhuber (1997) [26]. Unlike standard feedforward networks, RNNs are designed to process sequential data, using past information to inform current and future outputs. LSTMs enhance standard RNNs by introducing a 'memory cell' regulated by three complex 'gates': a forget gate (determining what information to discard), an input gate (determining what new information to store), and an output gate (determining what information from the cell to output) [27]. This mechanism allows the model to intelligently retain relevant information over extended time horizons (e.g., annual seasonal patterns) while forgetting irrelevant short-term noise [28]. The primary advantage of LSTM is its ability to model complex non-linear patterns and long-term dependencies simultaneously.

Real-world time-series data, such as financial market data or search interest, are rarely linear and are often characterized by shifting trends and sudden shocks [22].

Forecasting has traditionally relied on conventional linear statistical models, such as ARIMA or SARIMA, to capture shifting trends and other predictive factors [8], [29]. However, these conventional models fail to capture complex non-linear patterns driven by dynamic events. Consequently, Deep Learning techniques like Gated Recurrent Units (GRU) and Convolutional Neural Networks (CNN) have been adopted to capture the non-linear intricacies that traditional statistics miss [11], [13]. Unlike CNNs, which focus on local features [30], or GRUs, which simplify memory mechanisms for the sake of speed [31], LSTM possesses a superior capacity to preserve long-term dependencies without experiencing information degradation. This advantage renders LSTM more precise in capturing long historical contexts, making it crucial for predicting the volatility of complex financial data [32]. LSTM has proven in various sequential forecasting applications, including financial market prediction, e-commerce demand forecasting [33], and risk forecasting [32].

Nevertheless, despite their superior accuracy, these deep learning models are inherently deterministic. They produce only a single-point forecast. In reality, financial managers, particularly within the pawning industry, require more than just high-precision point predictions; they critically need measurable risk scenarios. These probabilistic scenarios are essential for effectively allocating liquidity reserves and formulating strategic defenses against sudden demand surges. This specific gap is addressed in this study through the hybridization of LSTM with Monte Carlo (MC) Dropout.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The "gadai" Google Trends data is ideal for LSTM due to its non-linear (event-driven) and seasonal (cycle-driven) nature. LSTM is capable of mapping these complex patterns to generate an accurate baseline forecast. However, a standard LSTM model is deterministic: it produces only a point forecast (a single-number prediction). This is a significant methodological weakness as it fails to quantify the uncertainty of its prediction—a limitation addressed by the next hybrid component..

2.2. Monte Carlo

IT Monte Carlo (MC) Simulation, also known as the Monte Carlo Method, is a mathematical computation technique used to estimate the possible outcomes of an uncertain event. Invented by John von Neumann and Stanislaw Ulam during World War II to model probabilistic scenarios, it was named after the casino in Monaco, highlighting the element of chance at the core of its approach [34]. Unlike deterministic models, MC runs thousands (or millions) of 'scenario' trials using random sampling from a probability distribution [35], [36].

This technique has demonstrated significant efficacy across various domains, particularly in production, politics and medical diagnostics [37]–[40]. While its adoption within the broader financial sector is growing, its application specifically within the pawning industry remains scarce. Although previous literature has explored hybridizing this method with other techniques, none have specifically applied it to address liquidity risks in pawning [41], [42]. Consequently, this study extends the application of MC Dropout beyond a mere technical optimization tool, repositioning it as a strategic instrument for risk mitigation in the pawning industry. Its greatest strength is its ability to transform uncertainty into quantifiable risk. Monte Carlo has become a cornerstone of modern risk analysis, particularly in finance for calculating Value-at-Risk (VaR), pricing options, and evaluating investments [38], [43].

This research proposes a hybrid model to overcome the point-forecast limitation of LSTM. In this framework, the LSTM is tasked with finding the complex 'signal' (non-linear patterns) from the historical data. Subsequently, Monte Carlo (implemented via the MC Dropout technique) is tasked with simulating the 'noise' (uncertainty/risk) around that signal. This combination allows us to shift from the point forecast (provided by LSTM) to a probabilistic forecast (provided by MC), directly addressing the need for managerial risk scenarios.

2.3. Google Trends

Google Trends (GT) functions as a real-time index of public search interest. The data is presented on a scale of 0 to 100, where 100 signifies peak relative popularity for a specified query, region, and timeframe. Due to its real-time and high-frequency nature, GT data has become a subject of intensive academic research [44].

A key advantage of GT data is its role as a leading indicator of human behavior, contrasting sharply with traditional economic data, which are often lagging indicators [4], [5].

In the fields of business and economics, GT data has been effectively used for nowcasting (predicting the present) and forecasting various variables. Established applications include forecasting retail sales, predicting stock market prices [6] , estimating fashion product choices [4] and monitoring unemployment rate [7].

However, the utilization of Google Trends data as a proxy for pawnshop (gadai) demand remains significantly underexplored.

3. METHODOLOGY

This chapter details the methodological framework used to answer the research question. This study adopts a hybrid time-series forecasting framework that integrates deep learning (LSTM) for pattern mapping and stochastic simulation (Monte Carlo) for risk quantification. The research workflow is visualized in Figure 1.

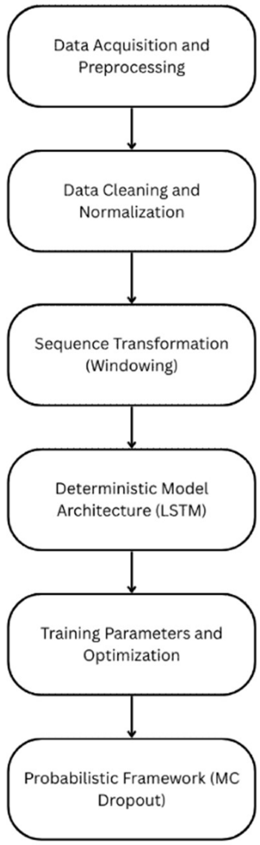

Figure 1: Research Methodology Flowchart

Figure 1 represent the research methodology in this study which consists of five primary stages:

3.1. Data Acquisition and Preprocessing

This stage involves data retrieval and preparation. The dataset used is the Google Trends time-series for the keyword "gadai" in Indonesia (geo='ID'). The dataset comprises 5 years of daily data (November 1, 2020, to October 31, 2025), acquired on November 12, 2025, totaling approximately 1,826 data points. As the Google Trends API restricts daily data retrieval to periods of less than 9 months, we implemented a "rescaling overlapping-chunks" method. Data was retrieved in 8-month segments with a 1-month overlap. Each new segment was then mathematically joined to the previous segment to create a single, consistent, and complete 5-year daily time-series.

1. Data Cleaning and Normalization

After data acquisition, the following steps were taken to ensure data quality and readiness for the LSTM model:

- Handling Missing Data. This stage addresses missing values. Sporadic missing (NaN) or zero (0) values within the Google Trends data were handled using time-based linear interpolation (.interpolate(method='time')) [45], [46].
- Data Splitting. The dataset was chronologically split into an 80% training set and a 20% testing set. This split is essential to simulate a real-world scenario where the model is trained on the past to predict an unseen future.
- Data Scaling and Leakage Prevention. This stage scales the data, as LSTM models are highly sensitive to the input scale. We utilized the MinMaxScaler from scikit-learn to normalize all data to a [0, 1] range. A critical justification for this step is the prevention of data leakage: the MinMaxScaler is fit (i.e., learns the min/max values) only on the training set. This same fitted scaler is then used to .transform() both the training and testing sets, ensuring no information from the future (the test set) "leaks" into the training process [47], [48].

3.2. Sequence Transformation (Windowing)

This stage is the transformation step for the LSTM. An LSTM model does not predict from a single data point, but from a sequence. This research converts the normalized univariate series into a supervised learning format using a sliding window method, with the following steps:

- Timestep (Look-back). LSTM learns from time-series data, so this part sets the duration for supervised learning. Based on initial experiments, the TIMESTEP was set to 60 days.
- Process. The model is given an input (X) comprising a sequence of 60 days of historical data to predict the output (y) on the 61st day. This window is then slid forward one day (stride=1) across the entire dataset.
- Reshaping. The resulting X_train and X_test inputs are then reshaped into the 3D tensor format required by Keras-LSTM: [Samples, Timesteps, Features]. For this study, the shape is (N, 60, 1).

3.3. Hybrid Model Architecture (LSTM-MC) - Deterministic Model Architecture (LSTM)

The core methodological foundation of this research is a hybrid framework designed to achieve two primary objectives, first, to map non-linear patterns using Long Short-Term Memory (LSTM), and second, to quantify risk using Monte Carlo simulation. These objectives are achieved first through the LSTM methodology. Following a series of iterative experiments to acquire clean and processed data, the subsequent analysis was conducted using a model architecture built with the Sequential API from Keras from TensorFlow [49], [50]. The LSTM architecture was designed as a layered analytical workflow [51], [52], detailed as follows:

- Layer 1 (Input/Pattern Mapping): The first LSTM layer with 100 units (neurons). This layer receives the input sequence (TIMESTEP=60) and is responsible for mapping the initial temporal patterns. The return_sequences=True parameter is activated to ensure the layer passes the full sequence of findings (rather than just the final output) to the subsequent layer.
- Layer 2 (Regularization): This layer randomly deactivates 20% of neurons during the training phase by setting Dropout(rate=0.2). This is a critical regularization mechanism to prevent overfitting by ensuring the model does not become overly reliant on individual neurons for feature detection.
- Layer 3 (Pattern Synthesis): The second LSTM layer, also with 100 units. This layer receives the sequence of findings from the first layer and synthesizes them into a single final representation vector. The return sequences=False setting instructs this layer to return only the output from the final timestep.
- Layer 4 (Regularization): A second Dropout(rate=0.2) layer is applied to regularize the output from the synthesis layer and further mitigate the risk of overfitting.
- Layer 5 (Output): This layer functions as the final decision-maker, translating the complex synthesis vector from the LSTM layers into a single scaled prediction value (between 0 and 1). This is implemented as a standard Dense layer with 1 unit and a linear activation function.

3.4. Training Parameters and Optimization.

The architecture described above is the "blueprint"; the training process is key to obtaining a stable and generalizable model. Based on empirical observations during our iterative process, the following training parameters were specifically chosen to stabilize the model:

- Optimizer (Learning Method): This research utilizes the Adam optimizer. Initial observations revealed that the default learning rate (LR) of 0.001 was too high, causing a highly unstable or "noisy" validation loss curve. Therefore, we explicitly defined the Adam optimizer with a lower custom learning rate of learning_rate=0.0001. This step proved crucial in producing the smooth and convergent learning curve essential for a reliable model.
- Batch Size (Learning Group): Set to 32. This smaller batch size (compared to the common default of 64) was chosen to facilitate a more frequent and smoother weight update process. This decision provably contributed to the stability of the validation loss during training.
- Loss Function: We utilized mean_squared_error (MSE) as the loss function, which is the standard metric for regression-based forecasting problems [33], [53].
- Overfitting Prevention (Model Auditor): In addition to Dropout, we implemented the EarlyStopping callback. This mechanism monitors the val_loss metric (the model's performance on unseen test data) at each epoch. Training was configured to halt automatically if the val_loss did not show improvement for a patience=15 epoch window. Critically, we used restore_best_weights=True, ensuring that the final model employed in this study is the one from the epoch with the lowest val_loss, not the model from the final epoch which may have already begun to overfit.

3.5. Probabilistic Framework (MC Dropout)

The model trained in the preceding steps is deterministic; it produces a single "best guess" (point forecast). To transition to probabilistic forecasting, this study adopts the Monte Carlo (MC) Dropout technique, which was theoretically validated by Gal & Ghahramani (2016) [31]. This method interprets dropout as a Bayesian approximation for uncertainty quantification [54].

- Implementation: We take the trained LSTM model (retrieved from the best epoch by EarlyStopping) and call it N_SIMULATIONS = 1000 times to predict the same data points.
- Core Mechanism: In each of these 1,000 simulation calls, we explicitly set the training=True parameter. This forces the Dropout(0.2) layers to remain active during the prediction (inference) phase.
- Output (Distribution): Because the neurons "deactivated" by the dropout layers are random in every call, each of the 1,000 predictions yields a slightly different output. This collection of 1,000 diverse predictions forms an empirical probability distribution (histogram) of the expected output. This final distribution allows to measure risk and calculate confidence intervals.

4. RESULTS AND DISCUSSIONS

This section reports and analyzes the results of the hybrid modeling performed on the "Gadai" (Pawning) Google Trends data. The reporting in this chapter is divided into five stages that narrate the model's proof-of-concept.

4.1. Descriptive Statistics of "Gadai" Data

Prior to predictive modeling with Deep Learning, a descriptive analysis was conducted on the daily "Gadai" Google Trends time-series dataset (N=1827 days). This step is crucial for understanding the data's characteristics and distribution. The descriptive results are presented in **Table 1**.

Table 1: Descriptive Statistics of Daily "Gadai" Search Interest

Statistic	Value
Count (N)	1827
Mean	79.794
Std. Deviation	**17.794**
Min	26.765
25% (Q1)	67.000
50% (Median)	78.176
75% (Q3)	89.930
Max	143.798

Based on Table 1, the mean daily search index was 79.79. The wide disparity between the minimum (26.76) and maximum (143.80) values, as well as the significant standard deviation (17.794), confirms that the data possesses high volatility and non-linear fluctuations. This unstable data characteristic is ideal for an LSTM-based approach, which can capture dynamic patterns more effectively than traditional linear statistical methods (such as ARIMA).

4.2. Model Training Dynamics and Validation

The LSTM model was configured using a *stacked* architecture as detailed in Chapter 3 (comprising two 100-unit LSTM layers and two 0.2-rate Dropout layers). The training process was conducted for 100 epochs, utilizing an EarlyStopping mechanism and the optimized parameters (lr=0.0001, batch_size=32).

The training results (summarized from STAGE 7 logs) show that the *Loss function* (MSE) on both the training data (Training Loss) and validation data (Validation Loss) converged in a stable manner. This stability is confirmed by the validation loss, which moved harmoniously around 0.0179 at the end of training. [**Note:** The training was automatically halted by EarlyStopping at **Epoch 72** (as per log file), restoring the best weights from the lowest point of the val_loss curve.] This result indicates stable convergence, suggesting the LSTM model successfully learned the sequential patterns of the data without succumbing to severe overfitting [53]. The results of this training are visualized in Figure 2.

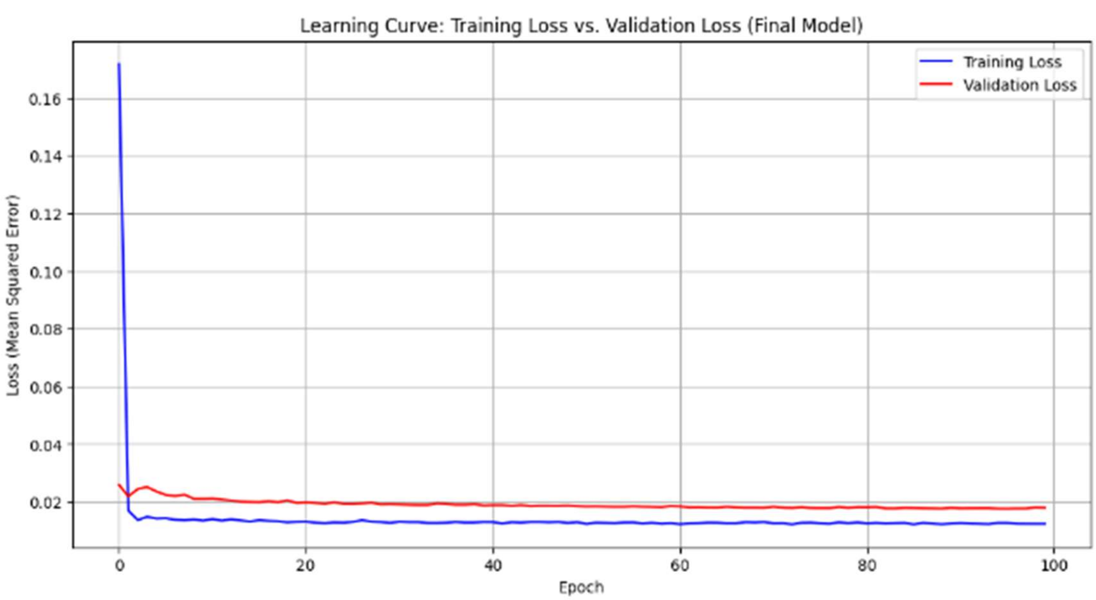

Figure 2: Training and Validation Phases

Figure 2 visualizes the descent trajectory of the *Loss* (MSE) value over the training iterations (epochs). The blue curve represents the error on the training data (*Training Loss*), while the red curve represents the error on the validation data (*Validation Loss*). Based on this graph, three diagnostic conclusions can be drawn:

1. Rapid Convergence: A sharp decrease in both Training and Validation Loss occurs within the first 5 epochs (dropping from ~0.14 to ~0.02). This indicates that the Adam optimizer with the custom learning_rate=0.0001 was highly effective in finding the proper gradient descent at the beginning of the process.
2. Generalization Capability: The minimal gap between the blue and red lines, and the fact that both curves move in tandem (converge) until the end, proves that the model possesses excellent generalization capability (*robustness*). This phenomenon demonstrates that the applied Dropout(0.2) and EarlyStopping mechanisms successfully prevented the model from becoming overly complex and overfitting.
3. Asymptotic Stability: From approximately epoch 20 to the end, both curves plateau. This indicates the model had reached its optimal performance point. EarlyStopping with patience=15 precisely identified this saturation point, making the decision to halt training computationally efficient.

4.3. Deterministic Model Performance Evaluation

This section validates the model's fundamental ability to capture data trend patterns (*point forecast*) and ensures the prediction error rate falls within an acceptable tolerance (<10%).

In the initial evaluation phase, the model was tested deterministically (without activating MC Dropout during inference). The model's performance was measured using Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) as standard metrics. The results are presented in Table 2.

Table 2: Model Performance Evaluation Metrics

Metric	Training Set	Test Set
RMSE	8.547	10.644
MAPE (%)	9.57%	7.90%

Based on Table 2, the model achieved a MAPE of 7.90% on the Test Set. Referencing the forecasting accuracy standard [55] , a MAPE value below 10% can be categorized as "Highly Accurate" prediction capability. The RMSE value of 10.644 implies that the average deviation of predictions from the actual values is approximately 10.6 index units, which is within operational tolerance limits [56].

This performance is visually supported by Figure 3.

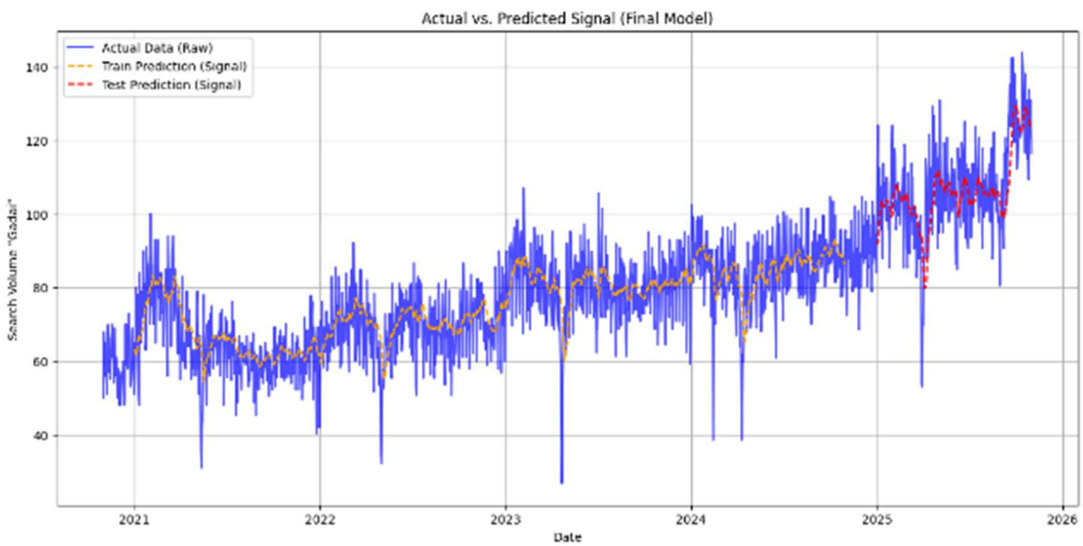

Figure 3:Comparison of Actual Signal vs. Model Prediction

The visualization in Figure 3, which superimposes the actual data (blue line) with the model's predictions (dashed orange/red lines), confirms three crucial findings about the constructed LSTM model:

1. Sensitivity to Trend Changes: The prediction line (especially the red test-set line) demonstrates an ability to adapt and follow the "uptrend" of the actual data, particularly in the 2024-2025 period, without significant *lag*. This proves the model did not just memorize past

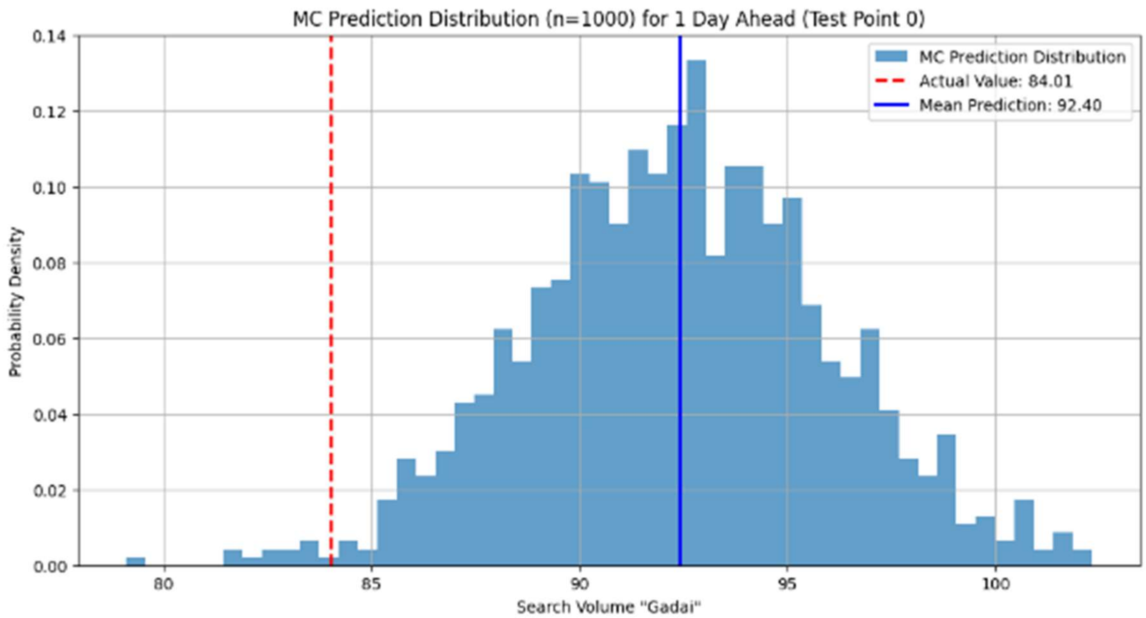

Figure 4: Monte Carlo Prediction Distribution (n=1000) for Test Point 0

patterns but could generalize to new dynamic data.

2. Noise Filtering Effect: The prediction lines (orange and red) appear significantly "smoother" than the highly *spiky* (jagged) actual data line. This is an ideal outcome. The model has successfully separated the "Underlying Signal" from the "Daily Noise." By producing a smoother curve, the model provides a more *robust* trend estimate for managerial decision-making, avoiding overreaction to insubstantial daily fluctuations.
3. Cross-Phase Consistency: There is no discernible performance discrepancy between the training phase (orange) and the testing phase (red); both adhere closely to the actual data. This confirms the model is not *overfitting* (memorizing noise). The model successfully extracted fundamental patterns from the training data and applied them to unseen test data, demonstrating high *robustness* and *generalizability*.

4.4. Probabilistic Analysis and Uncertainty Quantification

This stage represents the primary contribution (*novelty*) of this research, where we implement Monte Carlo (MC) Dropout to capture *epistemic uncertainty*. By running 1,000 simulations for each test point, the model generates a probability distribution of predictions, rather than a single point estimate. The results are presented in Figure 4 and Table 3.

Table 3: Probabilistic Risk Analysis Results (Sample of First 10 Test Days) (Data from STAGE 10 Script)

Test Day	Actual	MC Mean	95% CI Lower	95% CI Upper	Status
0	84.01	92.62	85.30	99.94	**Anomaly**
1	99.53	91.93	84.85	98.92	**Anomaly**
2	115.04	92.28	84.39	99.32	**Anomaly**
3	124.08	94.52	86.80	101.87	**Anomaly**
4	96.94	97.78	90.34	105.49	Validated
5	108.57	98.98	90.63	107.24	**Anomaly**
6	112.45	100.58	92.36	108.80	**Anomaly**
7	102.11	102.39	93.84	110.18	Validated
8	107.28	102.81	94.04	110.86	Validated
9	95.65	103.76	94.75	112.23	Validated

Figure 4 and Table 3 visualizes the core output of the hybrid model: a probability distribution from 1,000 scenarios for Test Day 0. The model's "best guess" (MC Mean) is 92.62, with a 95% Confidence Interval (CI) between 85.30 and 99.94.

However, the actual observed value (dashed red line) was 84.01. This is a key finding: the actual value fell outside the 95% CI. This is not a model failure; rather, it is evidence of the probabilistic method's *learning process*. The model, trained on 5 years of history, identified the event on Test Day 0 as a statistical anomaly (a <2.5% probability event). A standard LSTM would merely predict 92.62 and be marked as "wrong." The hybrid LSTM-MC, however, provides crucial context: "The model predicts 92.62, *and* it signals that the actual event of 84.01 is a 'surprise' in the market."

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 4.3 shows this pattern continued for the first four days (labeled "Anomaly") before the market returned to "normal" behavior (labeled "Validated"). This demonstrates a shift from mere "accuracy" to **"**risk management**."** For a marketing manager in the pawning industry, this output is an early warning indicator. When the actual value falls outside the predicted range, it triggers important managerial questions about unexpected market events. The model successfully quantifies uncertainty and transforms data into actionable insight.

4.5. Model Comparison: Deterministic vs. Probabilistic

To validate the effectiveness of the probabilistic approach, a head-to-head comparison was conducted between the standard deterministic LSTM prediction and the mean of the 1,000 MC simulations as shown in Table 4.

Table 4: Model Performance Comparison (Mean Accuracy)

Model Type	Test RMSE	Test MAPE (%)
LSTM (Deterministic)	10.644	7.90%
LSTM-MC (Probabilistic Mean)	10.646	7.89%

The results in Table 4 are highly significant. The RMSE and MAPE figures for both models are nearly identical. This proves that the addition of the probabilistic layer (MC Dropout) does not sacrifice or degrade the mean predictive accuracy. Instead, the MC Dropout method provides massive added value (i.e., "risk information" and confidence intervals) without compromising core forecasting precision.

4.6. Comparative Analysis

Based on the empirical evaluation, the proposed hybrid model successfully demonstrated high predictive performance. However, the primary significance of this study extends beyond mere accuracy metrics. The core novelty lies in the paradigm shift it offers: transitioning from static, deterministic forecasting to dynamic, probabilistic risk assessment.

Conventional linear models, such as ARIMA and ARIMAX, have proven empirically effective for forecasting stable macro-trends, such as epidemics or unemployment rates [7], [57]. However, these traditional models are inadequate for capturing the complex non-linear patterns and sudden shocks that characterize distress financing and pawning demand. Addressing this limitation represents one of the distinct advantages of the proposed hybrid framework.

Furthermore, to address non-linearity, recent literature has advanced towards Deep Learning architectures. Studies utilizing hybrid SARIMA-LSTM for aerosol data [9], Bidirectional CNN-LSTM for stock prices [22], or other CNN-based models [13] have indeed significantly improved predictive precision. Nevertheless, these studies share a fundamental limitation: they remain deterministic. They generate only single-point forecasts without indicating the model's confidence level—a critical gap for effective management decision-making.

Even when compared to the "closest" existing work, specifically studies forecasting global drought using similar Google Trends and LSTM [15], this research offers a distinct value-add. While such studies validate the power of Google Trends as an early signal, they are limited to single-point value predictions without quantifying the dimension of uncertainty. Conversely, this research fills that gap by introducing confidence intervals through the Monte Carlo probabilistic layer. Thus, this study completes the literary evolution: moving from rigid linear statistics and risk-blind deterministic Deep Learning, toward a Hybrid LSTM-Monte Carlo Dropout model capable of quantifying previously neglected epistemic uncertainty. This provides pawning managers with a strategic instrument to map "upper bound risk" amidst market volatility.

5. CONCLUSIONS

The primary objective of this research was to demonstrate the efficacy of a hybrid LSTM-Monte Carlo model in predicting public interest trends in Indonesian pawning services using Google Trends data, while simultaneously providing valid probabilistic risk assessments. Utilizing a dataset spanning 5 years (November 1, 2020, to October 31, 2025; N=1,826), this study employed a deterministic LSTM model integrated with Monte Carlo (MC) Dropout. The empirical results confirm the robustness of the proposed framework. The deterministic LSTM component achieved high predictive accuracy with a MAPE of 7.90%, classifying it as a "Highly Accurate" forecasting model. The LSTM architecture successfully extracted fundamental signals from highly volatile search data, effectively filtering out insubstantial daily noise, a capability that overcomes the limitations of traditional linear models which often overfit to noise. Furthermore, the probabilistic component (MC Dropout) successfully transformed single-point predictions into risk distributions. The model demonstrated its capability as an anomaly detector, identifying "market surprises" (e.g., on Test Days 0–3) where actual values fell outside the 95% Confidence Interval. These findings indicate that the model functions not only as a predictor but also as a robust risk management tool.

Based on the achievement of the initial research objectives, the authors conclude that the contribution of this study extends beyond mere accuracy metrics. Fundamentally, the hybridization of LSTM with MC Dropout offers a paradigm shift in Deep Learning application: moving from rigid single-point predictions to dynamic risk-distribution forecasting. Specifically for the pawning industry, this finding transforms liquidity management from reactive demand estimation into an anticipatory and measurable risk mitigation instrument.

6. IMPLICATIONS

This study makes significant contributions to the existing literature in three key areas:

Time-Series Forecasting Literature: We demonstrate that integrating deep learning (LSTM) with stochastic simulation (Monte Carlo) can be achieved without sacrificing mean predictive accuracy (Probabilistic MAPE 7.89% vs. Deterministic MAPE 7.90%). This finding validates that uncertainty quantification adds massive value without degrading core forecasting performance.

Pawning Industry Literature: This study contributes by constructing a specialized predictive model for the Indonesian pawning sector. It validates the use of Google Trends as a reliable leading indicator for pawning demand, filling a gap in digital financial behavior research in developing markets.

Risk Management Literature: We contribute a novel framework for "Digital Risk Sensing". By applying MC Dropout to search query data, we provide a method to quantify the epistemic uncertainty of market interest, offering a new metric for assessing financial distress risks in real-time.

Beyond theoretical advancements, this study offers actionable implications for practitioners in the pawning and microfinance industry:

Early Warning System: The model serves as a real-time early warning system. When realized demand falls outside the predicted 95% Confidence Interval, managers can immediately trigger response protocols, such as investigating external market anomalies or adjusting liquidity positions.

Operational Risk Management: The probabilistic distribution allows managers to shift from "average-based planning" to "risk-based planning." For instance, branch managers can optimize cash reserves based on the upper bound of the 95% CI (worst-case demand scenario) rather than the mean, thereby minimizing the risk of liquidity stockouts while maintaining efficiency.

7. LIMITATIONS AND SUGGESTIONS

This study acknowledges two primary limitations. First, from a methodological perspective, the current model functions as a single step forecaster (predicting only 1 day ahead); consequently, its capability for multi-step forecasting and the potential impact of compounding errors over long-term horizons remain untested. Second, regarding data validity, the study utilizes Google Trends which serves as a proxy for public interest rather than actual finalized transaction data.

To address these constraints, future research is suggested to: (1) Explore multi-step architectures, such as Encoder-Decoder LSTM, to enable medium-term forecasting (7–30 days); and (2) Integrate multivariate external variables, such as global gold prices, inflation rates, or exchange rates to enrich the context and enhance the robustness of the predictive model.

Data Availability Statement

The dataset generated and analyzed during the current study is openly available in the Figshare repository, e.g., https://doi.org/10.6084/m9.figshare.30663515.

Author Contributions

Wanda: Conceptualization, Methodology, Software, Formal analysis, Writing original draft, Visualization. Robert: Data curation, Investigation, Validation, Writing –review & editing. Enda: Supervision, Project administration, Writing review & editing.

Declaration of Generative AI and AI-assisted Technologies in the Writing Process

During the preparation of this work, the authors used Large Language Models (LLMs) to improve the readability, grammatical accuracy, and flow of the English language in the manuscript. After using this tool/service, the authors reviewed and edited the content and take full responsibility for the content of the publication. The AI tool was utilized strictly for language refinement and not for the generation of scientific concepts, data interpretation, or the formulation of the core conclusions.

Ethical Statement

This research utilizes aggregated and anonymized data retrieved from Google Trends, which falls under the category of non-interactive public data. As the dataset does not contain Personally Identifiable Information (PII) and involves no direct intervention with human subjects, formal ethical approval from an Institutional Review Board (IRB) was not required according to standard academic research guidelines.

REFERENCES

[1] R. Binekasri, "Bisnis Gadai Makin Disukai Warga RI," *cnbcindonesia.com*, 2025. [Online]. Available: https://www.cnbcindonesia.com/market/20251013111021-17-675238/bisnis-gadai-makin-disukai-warga-ri
[2] M. Y. Edward, E. N. Fuad, H. Ismanto, A. D. R. Atahau, and R. Robiyanto, "Success factors for peer-to-peer lending for SMEs: Evidence from Indonesia," *Invest. Manag. Financ. Innov.*, vol. 20, no. 2, pp. 16–25, 2023, doi: 10.21511/imfi.20(2).2023.02.
[3] S. Carta, A. Medda, A. Pili, D. R. Recupero, and R. Saia, "Forecasting E-commerce products prices by combining an autoregressive integrated moving average (ARIMA) model and Google Trends data," *Futur. Internet*, vol. 11, no. 1, 2018, doi: 10.3390/fi11010005.
[4] G. Skenderi, C. Joppi, M. Denitto, and M. Cristani, "Well googled is half done: Multimodal forecasting of new fashion product sales with image-based google trends," *J. Forecast.*, vol. 43, no. 6, pp. 1982–1997, 2024, doi: 10.1002/for.3104.
[5] M. Gholamzadeh, M. Asadi Gharabaghi, and H. Abtahi, "Public interest in online searching of asthma information: insights from a Google trends analysis," *BMC Pulm. Med.*, vol. 25, no. 1, 2025, doi: 10.1186/s12890-025-03545-9.
[6] H.-C. Fan Chiang, P.-X. Jiang, and C.-C. Chang, "Google Search Trends and Exchange Rate Volatility – Evidence from India's Currency Market," pp. 195–210, 2021, doi: 10.1108/s2514-465020210000009010.
[7] W. K. Adu, P. Appiahene, and S. Afrifa, "VAR, ARIMAX and ARIMA models for nowcasting unemployment rate in Ghana using Google trends," *Journal of Electrical Systems and Information Technology*, vol. 10, no. 1. 2023. doi: 10.1186/s43067-023-00078-1.
[8] E. Kraft, D. Keles, and W. Fichtner, "Modeling of frequency containment reserve prices with econometrics and artificial intelligence," *J. Forecast.*, vol. 39, no. 8, pp. 1179–1197, 2020, doi: 10.1002/for.2693.
[9] N. K. K. Panicker and J. Valarmathi, "a Hybrid Sarima-Lstm Approach for Improved Time Series Prediction of Aerosol Optical Depth Across Delhi,India," *J. Theor. Appl. Inf. Technol.*, vol. 102, no. 11, pp. 4836–4853, 2024.
[10] X. Li, M. Khishe, and L. Qian, "Evolving deep gated recurrent unit using improved marine predator algorithm for profit prediction based on financial accounting information system," *Complex Intell. Syst.*, vol. 10, no. 1, pp. 595–611, 2024, doi: 10.1007/s40747-023-01183-4.
[11] M. Thor and Ł. Postek, "Gated recurrent unit network: A promising approach to corporate default prediction," *J. Forecast.*, vol. 43, no. 5, pp. 1131–1152, 2024, doi: 10.1002/for.3057.
[12] D. M. Durairaj and B. H. K. Mohan, "A convolutional neural network based approach to financial time series prediction," *Neural Comput. Appl.*, vol. 34, no. 16, pp. 13319–13337, 2022, doi: 10.1007/s00521-022-07143-2.
[13] J. Guo, W. Kang, and Y. Wang, "Multi-perspective option price forecasting combining parametric and non-parametric pricing models with a new dynamic ensemble framework," *Technol. Forecast. Soc. Change*, vol. 204, 2024, doi: 10.1016/j.techfore.2024.123429.
[14] N. Golenvaux, P. G. Alvarez, H. S. Kiossou, and P. Schaus, "An LSTM approach to Forecast Migration using Google Trends," 2020, [Online]. Available: http://arxiv.org/abs/2005.09902
[15] S. M. B. Shahabi-Haghighi and H. Hamidifar, "Exploring the link between drought-related terms and public interests: Global insights from LSTM-based predictions and Google Trends analysis," *Hydrol. Process.*, vol. 37, no. 11, 2023, doi: 10.1002/hyp.15016.
[16] M. K. Khafidli and A. Choiruddin, "Forecast of Aviation Traffic in Indonesia Based on Google Trend and Macroeconomic Data using Long Short-Term Memory," *2022 Int. Conf. Data Sci. Its Appl. ICoDSA 2022*, pp. 220–225, 2022, doi: 10.1109/ICoDSA55874.2022.9862894.
[17] P. Dash, J. Mishra, and S. Dara, "Bidirectional Cnn-Lstm Architecture To Predict Cnxit Stock

Prices," *J. Theor. Appl. Inf. Technol.*, vol. 102, no. 5, pp. 1896–1907, 2024.

[18] I. Rahmatillah, I. Sudirman, A. M. Azis, and I. D. Sudirman, "Lstm Neural Networks With Bayesian Optimization for Reorder Simulation in Retail Inventory Management," *J. Theor. Appl. Inf. Technol.*, vol. 102, no. 22, pp. 8193–8194, 2024.

[19] L. L. Boaventura, R. L. Fiaccone, and P. H. Ferreira, "Prediction Control Charts: A New and Flexible Artificial Intelligence-Based Statistical Process Control Approach," *Ann. Data Sci.*, vol. 11, no. 1, pp. 273–306, 2024, doi: 10.1007/s40745-022-00441-5.

[20] X. Xie, B. Xie, J. Cheng, Q. Chu, and T. Dooling, "A simple Monte Carlo method for estimating the chance of a cyclone impact," *Nat. Hazards*, vol. 107, no. 3, pp. 2573–2582, 2021, doi: 10.1007/s11069-021-04505-2.

[21] A. Akins, D. Kultgen, X. Wu, and A. Heifetz, "Monte Carlo Dropout Uncertainty Quantification of Long Short-Term Memory Autoencoder Anomaly Detection in a Liquid Sodium Cold Trap," *Nucl. Technol.*, pp. 1–14, 2025, doi: 10.1080/00295450.2025.2518613.

[22] S. Srivastava, M. Pant, and V. Gupta, "Analysis and prediction of Indian stock market: a machine-learning approach," *Int. J. Syst. Assur. Eng. Manag.*, vol. 14, no. 4, pp. 1567–1585, 2023, doi: 10.1007/s13198-023-01934-z.

[23] J. M. Failing, J. Segarra-Tamarit, J. Cardo-Miota, and H. Beltran, "Deep learning-based prediction models for spot electricity market prices in the Spanish market," *Math. Comput. Simul.*, vol. 240, no. January 2025, pp. 96–104, 2026, doi: 10.1016/j.matcom.2025.07.010.

[24] M. Ievlanov, N. Vasyltsova, I. Panforova, and Y. Kliuvanskyi, "Improving a Risk Estimating Method for the 'Smart House' Information System It Project," *Eastern-European J. Enterp. Technol.*, vol. 1, no. 3(133), pp. 6–16, 2025, doi: 10.15587/1729-4061.2025.322051.

[25] J. Huang *et al.*, "Health risk assessment of heavy metal(loid)s in park soils of the largest megacity in China by using Monte Carlo simulation coupled with Positive matrix factorization model," *J. Hazard. Mater.*, vol. 415, no. February, p. 125629, 2021, doi: 10.1016/j.jhazmat.2021.125629.

[26] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[27] M. Z. Alom *et al.*, "A state-of-the-art survey on deep learning theory and architectures," *Electron.*, vol. 8, no. 3, 2019, doi: 10.3390/electronics8030292.

[28] F. J. Ordóñez and D. Roggen, "Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition," *Sensors (Switzerland)*, vol. 16, no. 1, 2016, doi: 10.3390/s16010115.

[29] G. Xu, T. Fan, Y. Zhao, W. Wu, and Y. Wang, "Predicting the epidemiological trend of acute hemorrhagic conjunctivitis in China using Bayesian structural time-series model," *Sci. Rep.*, vol. 14, no. 1, pp. 1–11, 2024, doi: 10.1038/s41598-024-68624-z.

[30] M.-C. Chiu and K.-H. Chuang, "Applying transfer learning to achieve precision marketing in an omni-channel system–a case study of a sharing kitchen platform," *Int. J. Prod. Res.*, vol. 59, no. 24, pp. 7594–7609, 2021, doi: 10.1080/00207543.2020.1868595.

[31] Y. Gal and Z. Ghahramani, "A theoretically grounded application of dropout in recurrent neural networks," *Adv. Neural Inf. Process. Syst.*, pp. 1027–1035, 2016.

[32] Y. Wang, H. Zhang, B. Huang, Z. Lin, and C. Pang, "LSTM stock prediction model based on blockchain," *High-Confidence Comput.*, vol. 5, no. 4, p. 100316, 2025, doi: 10.1016/j.hcc.2025.100316.

[33] L. Efrizoni, E. Ali, H. Asnal, and Junadhi, "Adaptive Neural Collaborative Filtering with Textual Review Integration for Enhanced User Experience in Digital Platforms," *J. Appl. Data Sci.*, vol. 6, no. 4, pp. 2696–2710, 2025, doi: 10.47738/jads.v6i4.944.

[34] A. I. A. Eid, W. Awang, Mzarina, and A. Zakaria, "Replication strategies based on markov chain monte carlo and optimization on cloud applications," *J. Theor. Appl. Inf. Technol.*, vol. 98, no. 3, pp. 517–534, 2020.

[35] S. Bonnini, M. Borghesi, and M. Giacalone, "Simultaneous marginal homogeneity versus directional alternatives for multivariate binary data with application to circular economy assessments," *Appl. Stoch. Model. Bus. Ind.*, vol. 40, no. 2, pp. 389–407, 2024, doi: 10.1002/asmb.2827.

[36] M. Ljungberg, "Introduction to the Monte Carlo Method," *Monte Carlo Calc. Nucl. Med. Appl. Diagnostic Imaging*, pp. 1–16, 2012, doi: 10.1201/b13073-6.

[37] N. N. Thach, "Applying Monte Carlo Simulations to a Small Data Analysis of a Case of Economic Growth in COVID-19 Times," *SAGE Open*, vol. 13, no. 2, pp. 1–13, 2023,

doi: 10.1177/21582440231181540.

[38] J. Fry, T. Hastings, and J. Binner, “An options-pricing approach to forecasting the French presidential election,” *J. Oper. Res. Soc.*, vol. 76, no. 1, pp. 167–179, 2025, doi: 10.1080/01605682.2024.2334339.

[39] X. Tao and D. Ucbasaran, “How does failure normalization foster product innovativeness in new product development? The role of passion and learning,” *J. Prod. Innov. Manag.*, vol. 42, no. 2, pp. 338–364, 2025, doi: 10.1111/jpim.12755.

[40] E. González-Núñez, L. A. Trejo, and M. Kampouridis, “A Comparative Study for Stock Market Forecast Based on a New Machine Learning Model,” *Big Data Cogn. Comput.*, vol. 8, no. 4, 2024, doi: 10.3390/bdcc8040034.

[41] A. Mohammed, M. A. A. Bakar, M. M. Mansor, and N. M. Ariff, “Modelling Malaysia Air Quality Data using Bayesian Structural Time Series Models,” *Sains Malaysiana*, vol. 53, no. 11, pp. 3817–3829, 2024, doi: 10.17576/jsm-2024-5311-23.

[42] J. Lee and T. Park, “Environmental factors, ambidexterity and performance in SMEs: does bricolage matter?,” *J. Bus. Ind. Mark.*, vol. 39, no. 3, pp. 521–536, 2024, doi: 10.1108/JBIM-07-2022-0336.

[43] D. P. Kroese, T. Brereton, T. Taimre, and Z. I. Botev, “Why the Monte Carlo method is so important today,” *Wiley Interdiscip. Rev. Comput. Stat.*, vol. 6, no. 6, pp. 386–392, 2014, doi: 10.1002/wics.1314.

[44] A. Zeller, “Google Trends Search Volume Explained,” *explodingtopics.com*, 2025. [Online]. Available: https://explodingtopics.com/blog/google-trends-search-volume

[45] N. Cesare and L. P. O. Were, “A multi-step approach to managing missing data in time and patient variant electronic health records,” *BMC Res. Notes*, vol. 15, no. 1, 2022, doi: 10.1186/s13104-022-05911-w.

[46] O. Ennasr *et al.*, “Characterization of acoustic detection efficiency using a gliding robotic fish as a mobile receiver platform,” *Anim. Biotelemetry*, vol. 8, no. 1, 2020, doi: 10.1186/s40317-020-00219-7.

[47] I. Izonin, R. Tkachenko, N. Shakhovska, B. Ilchyshyn, and K. K. Singh, “A Two-Step Data Normalization Approach for Improving Classification Accuracy in the Medical Diagnosis Domain,” *Mathematics*, vol. 10, no. 11, 2022, doi: 10.3390/math10111942.

[48] L. Bai, D. Mo, H. Li, W. Huang, and Z. Cai, “Optimized Data Preprocessing and Model Selection for TBM Cutter Wear Prediction,” *Coatings*, vol. 15, no. 5, 2025, doi: 10.3390/coatings15050564.

[49] K. Ramasubramanian and A. Singh, “Deep Learning Using Keras and TensorFlow,” *Mach. Learn. Using R*, pp. 667–688, 2019, doi: 10.1007/978-1-4842-4215-5_11.

[50] T. Lee, V. P. Singh, and K. H. Cho, “Tensorflow and Keras Programming for Deep Learning,” pp. 151–162, 2021, doi: 10.1007/978-3-030-64777-3_11.

[51] I. Chaturvedi, W. L. Seow, A. Hogarth, L. Adornetto, and E. Cambria, “Continuous Time Markov Chain for Smartwatch Sensors,” *Expert Syst.*, vol. 42, no. 11, pp. 1–9, 2025, doi: 10.1111/exsy.70144.

[52] V. Jain, L. Malviya, and A. .S, “Optimized hybrid deep learning for cross-linguistic sentiment analysis: a novel approach,” *J. Cloud Comput.*, vol. 14, no. 1, 2025, doi: 10.1186/s13677-025-00753-w.

[53] A. F. Aysan *et al.*, “A unified theory of acceptance and use of technology and fuzzy artificial intelligence model for electric vehicle demand analysis,” *Decis. Anal. J.*, vol. 11, no. January, p. 100455, 2024, doi: 10.1016/j.dajour.2024.100455.

[54] I. Alarab, S. Prakoonwit, and M. I. Nacer, “Illustrative Discussion of MC-Dropout in General Dataset: Uncertainty Estimation in Bitcoin,” *Neural Process. Lett.*, vol. 53, no. 2, pp. 1001–1011, 2021, doi: 10.1007/s11063-021-10424-x.

[55] J. Hameed, C. Huo, G. Albasher, and M. A. Naeem, “Revisiting the nexus between financialization and natural Resource efficiency through the lens of financial development and green industrial optimization,” *J. Clean. Prod.*, vol. 468, 2024, doi: 10.1016/j.jclepro.2024.143066.

[56] M. K. Nallakaruppan *et al.*, “Credit Risk Assessment and Financial Decision Support Using Explainable Artificial Intelligence,” *Risks*, vol. 12, no. 10, 2024, doi: 10.3390/risks12100164.

[57] F. Zhang, Y. Li, X. Li, B. Zhang, C. Xue, and Y. Wang, “Comparison of ARIMA and Bayesian Structural Time Series Models for Predicting the Trend of Syphilis Epidemic in Jiangsu Province,” *Infect. Drug Resist.*, vol. 17, no. December, pp. 5745–5754, 2024, doi: 10.2147/IDR.S462998.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195