BAREKENG: Journal of Mathematics and Its Applications
December 2026 Volume 20 Issue 4 Page 2711-2726
P-ISSN: 1978-7227 E-ISSN: 2615-3017
https://doi.org/10.30598/barekengvol20iss4pp2711-2726

COMPARING STATISTICAL AND DEEP LEARNING METHODS FOR INSURANCE CLAIM ESTIMATION: A CASE STUDY OF HIDDEN MARKOV MODEL (HMM) AND CONVOLUTIONAL NEURAL NETWORK - LONG SHORTTERM MEMORY (CNN-LSTM)

Ainun Mawaddah Abdal[1*], Andi Muhammad Anwar[2], Illuminata Wynnie[3], Amil Siddik[4], Edy Saputra Rusdi[5], Mauliddin[6]

[1,2,3,4,5,6]*Mathematics Department, Faculty of Mathematics and Natural Sciences, Universitas Hasanuddin*
Jln Perintis Kemerdekaan Km 10, Makassar, 92045, Indonesia.

*Corresponding author's e-mail: * **ainunabdal@unhas.ac.id***

Article Info

Article History:

Received: 11th Agustus 2025
Revised:12th May 2026
Accepted: 3rd June 2026
Published: 1st July 2026

Keywords:

Hidden Markov Model;
insurance claim forecasting;
Long Short-Term Memory.

ABSTRACT

The insurance industry relies heavily on accurate claim prediction to support risk management, reserve allocation, and financial stability. However, motor vehicle insurance claim data are typically characterized by temporal dependency, highly skewed distributions, and fluctuating claim severity, making accurate prediction a challenging task. While deep learning approaches have recently gained attention for time-series forecasting, their effectiveness on moderate-scale insurance claim datasets remains uncertain. This study aims to compare the predictive performance of the Hidden Markov Model (HMM) and CNN-LSTM in modelling temporal patterns and predicting daily motor vehicle insurance claims. In addition, an Attention-LSTM + XGBoost ensemble model is included as a supplementary deep learning benchmark. This study utilizes historical motor vehicle insurance claim data collected from 2017 to 2021, consisting of 11,679 claim observations. The data preprocessing stage included data cleaning, missing value handling, outlier detection, and claim severity categorization for HMM modelling. The HMM parameters were estimated using the Baum–Welch algorithm, while the deep learning models were trained using sequential claim data. Model performance was evaluated using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and coefficient of determination (R^2) on the testing dataset to ensure objective comparison. The experimental results show that the HMM model achieved the best predictive performance, outperforming both the CNN-LSTM and Attention-LSTM + XGBoost models. The findings indicate that the probabilistic structure of HMM is more suitable for modelling the temporal risk patterns and fluctuating claim behavior observed in the motor vehicle insurance dataset. Furthermore, the study demonstrates that classical probabilistic models can remain competitive and even outperform more complex deep learning approaches when applied to moderately sized insurance claim datasets with limited hidden complexity.

How to cite this article:

A. M. Abdal, A. M. Anwar, I. Wynnie, A. Siddik, E. S. Rusdi and Mauliddin., "COMPARING STATISTICAL AND DEEP LEARNING METHODS FOR INSURANCE CLAIM ESTIMATION: A CASE STUDY OF HIDDEN MARKOV MODEL (HMM) AND CONVOLUTIONAL NEURAL NETWORK -LONG SHORT TERM MEMORY (CNN-LSTM)", *BAREKENG: J. Math. & App.*, vol. 20, no. 4, pp. 2711-2726, Dec, 2026.

Journal homepage: *https://ojs3.unpatti.ac.id/index.php/barekeng/*
Journal e-mail: *barekeng.math@yahoo.com; barekeng.journal@mail.unpatti.ac.id*

Research Article · Open Access

1. INTRODUCTION

The insurance industry represents a strategic sector that fundamentally depends on effective long-term risk management. A key component of this management lies in the accurate prediction of claim frequency and severity. Reliable claim forecasting enables insurers to set actuarially sound premium rates, establish optimal claim reserves, and ensure financial stability. Conversely, inaccuracies in claim prediction may result in over-reserving, leading to inefficient capital allocation and diminished profitability, or under-reserving, which poses significant risks of claim payment shortfalls, liquidity crises, and potential insolvency [1], [2].

Empirical evidence indicates that Indonesia has experienced several high-profile insurance company bankruptcies over the past decade. According to a report by Lifepal (2023), major insurers such as Jiwasraya, AJB Bumiputera 1912, WanaArtha Life, Kresna Life, and Bakrie Life have failed to fulfil claim obligations amounting to trillions of rupiah. The Jiwasraya case, for instance, involved claim defaults totaling IDR 12.4 trillion, primarily due to mismanagement of insurance products and investment portfolios. WanaArtha Life reported a Risk-Based Capital (RBC) ratio of -2,018.53%, significantly below the minimum threshold set by the Financial Services Authority (OJK), highlighting a systemic inability to meet policyholder obligations. A common underlying factor in these failures is the absence of predictive and probabilistic data-driven risk management frameworks [3].

In line with technological advancements, claim prediction methodologies have evolved from traditional statistical approaches toward more adaptive and data-driven techniques, including stochastic modeling, deep learning, and ensemble learning methods. Insurance claim data commonly exhibit temporal dependencies, nonlinear fluctuations, volatility clustering, and hidden risk transitions, where current claim conditions are influenced by previous claim histories and underlying risk conditions [4]. Therefore, sequential prediction approaches such as Hidden Markov Models (HMM), Long Short-Term Memory (LSTM), CNN-LSTM architectures, and Attention-LSTM + XGBoost ensemble models have become increasingly relevant for insurance forecasting tasks.

In the study conducted by Jatipaningrum et al. [5], the application of the Hidden Markov Model (HMM) demonstrated high predictive accuracy for the Rupiah–US Dollar exchange rate, achieving a remarkably low error rate of less than 1.4%. The stability of the temporal pattern in the predicted results was attributed to the implementation of the Baum-Welch algorithm during the model training phase. Furthermore, the effectiveness of HMM in modeling stochastic time series data was also validated by Alimansyah and Purqon [6], who used the model for stock price prediction and reported an average prediction error of 27.04%, with the predicted trend closely aligning with the actual data.

Moreover, Long Short-Term Memory (LSTM), a variant of Recurrent Neural Networks (RNN), has proven effective in capturing long-term dependencies within complex sequential data. Sunny et al. [7] demonstrated the efficacy of LSTM in the context of financial forecasting by applying it to Google stock price prediction, where the model achieved an exceptionally low Root Mean Square Error (RMSE) of 0.0004928. This result indicates the model's superior predictive capability in handling time series data with high volatility and complexity. Furthermore, Cohen Sabban et al. [8] extended the application of LSTM to the insurance domain, where it was employed to classify severe insurance claims based solely on textual reports. To further enhance sequential prediction performance, hybrid architectures such as CNN-LSTM have recently been introduced to simultaneously capture local feature patterns and temporal dependencies within time-series data. CNN layers are capable of extracting short-term local patterns, while LSTM layers model long-term sequential behavior, making the architecture effective for complex forecasting problems [9].

In addition, ensemble learning approaches combining Attention-LSTM and XGBoost have attracted increasing attention due to their ability to improve predictive robustness and reduce model bias. The attention mechanism enables the model to focus on the most relevant temporal information within sequential data, while XGBoost enhances prediction stability through gradient boosting optimization. Recent studies have shown that attention-based ensemble models can outperform standalone deep learning architectures in various forecasting applications, including financial and risk prediction tasks [10]. These approaches are particularly promising for insurance claim prediction, where claim patterns may exhibit nonlinear fluctuations and varying temporal importance over time.

Although previous studies have demonstrated the effectiveness of HMM and deep learning approaches in modeling sequential data, most studies have focused on single-model implementations without conducting comprehensive comparative evaluations under the same insurance claim dataset. This creates a research gap,

particularly for motor vehicle insurance claim prediction, where the data are typically highly skewed, fluctuating, and characterized by occasional extreme claim events. Furthermore, the comparative effectiveness of classical probabilistic models and modern deep learning architectures under moderate-scale insurance datasets remains insufficiently explored.

Therefore, this study aims to compare the predictive performance of the Hidden Markov Model (HMM) and CNN-LSTM in modelling temporal patterns and predicting daily motor vehicle insurance claims. In addition, an Attention-LSTM + XGBoost ensemble model is included as a supplementary deep learning benchmark to further evaluate the effectiveness of advanced hybrid architectures for insurance claim prediction models.

2. RESEARCH METHODS

This study utilizes secondary data derived from historical motor vehicle insurance claim records from 2017 to 2021. Although insurance claims occur as event-based observations, the data were aggregated on a daily basis by summing the total claim amounts recorded on each calendar day. Daily aggregation was adopted to transform irregular claim events into a structured time-series format suitable for sequential modelling using HMM and CNN-LSTM, as commonly applied in insurance forecasting and actuarial analytics [2]. The dataset, stored in the Excel file "Claim KBM," contains transaction dates and corresponding claim amounts. To ensure reproducibility, the preprocessing procedures were conducted systematically, including duplicate removal, handling missing values through consistent imputation rules, and outlier detection using interquartile range (IQR)-based filtering.

For the HMM model, claim amounts were discretized into categorical states representing different claim severity levels to facilitate probabilistic state-transition modeling. The discretization thresholds were determined using data-driven quantile intervals to minimize arbitrary categorization and preserve the relative distribution of claim severity. Meanwhile, the original continuous numerical values were retained for CNN LSTM analysis to preserve complete quantitative information. This dual representation was intended to balance interpretability and predictive capability while reducing potential information loss caused by discretization. Previous studies have shown that quantile-based discretization can improve the stability of stochastic sequence models without introducing significant bias when the original data distribution is preserved appropriately [1].

Two modelling approaches are implemented: the Hidden Markov Model (HMM), trained using the Baum-Welch algorithm to predict claim categories and estimate amounts, and the Long Short-Term Memory (LSTM) network, trained on claim amount time series to capture temporal patterns. In addition, an ensemble learning approach combining Attention-based LSTM and Extreme Gradient Boosting (XGBoost) is employed to improve predictive accuracy by capturing important temporal dependencies, nonlinear relationships, and irregular fluctuations within the claim data. The prediction performance of all models is compared based on accuracy and error metrics, with all analyses conducted in Python using relevant data science libraries.

2.1 Hidden Markov Model

The Hidden Markov Model (HMM) is a statistical framework used to characterize the probability of different event sequences [11]. Since hidden states are unobservable, the Baum–Welch algorithm is applied to re-estimate model parameters. Wang et al. [12] demonstrated that HMM with Baum–Welch improves network risk assessment accuracy by inferring attack intentions.

In this model, the observation O_t at time t is generated by an underlying stochastic process, while the corresponding state S_t of this process is not directly observable, hence considered hidden. The hidden process adheres to the Markov property, implying that the current state S_t depends solely on the immediately preceding state S_{t-1} [13].

The steps for predicting claim amounts using the Hidden Markov Model (HMM) with the Baum–Welch algorithm are as follows [14], [15]:

1. Data Collection

   At this point, historical data in time series format is used, which includes details such as dates and claim severity.

$$Data = \{t_i, k_i\}, \tag{1}$$

   where t_i represents the i-th date, and k_i denotes the i-th claim severity.

2. Discretization of claim severity (Observation State)

   Claim severity is categorized by nominal value. The numerical *claim* data is transformed into three categories using quantiles: 0–25% (small), 25–75% (medium), and 75–100% (large). Mathematically, the claim category is denoted as

$$O_t \in \{1\ (Low), 2\ (Middle), 3(High)\}. \tag{2}$$

3. Initiation of the HMM

   a. Hidden State: The hidden states represent different risk levels associated with the claim amount,

$$S_t \in \{s_0, s_1, s_2), \tag{3}$$

   where s_0 corresponds to a normal risk level, s_1 indicates a medium risk level, and s_2 represents a highrisk level.

   b. Observation: The observations correspond to the claim categories

$$O_t \in \{o_0, o_1, o_2\}, \tag{4}$$

   using the same notation meaning at the discretization stage.

   c. HMM Parameters ($\lambda = (A, B, \pi)$):

      i. The transition matrix $A = \{a_{ij}\}$ is define such as that

$$a_{ij} = P(S_{t+1} = s_j | S_t = s_i), \tag{5}$$

      where a_{ij} represents the probability of transitioning from state s_i at time t to state s_j at time $t + 1$.

      ii. The emission matrix $B = \{b_j(k)\}$ is given by

$$b_j(k) = P(O_t = o_k | S_t = s_j), \tag{6}$$

      where $B = \{b_j(k)\}$ denotes the probability of observing o_k given that the system is in hidden state s_j, obtained by monitoring changes in the observed states.

      iii. The initial state probability vector $\pi = \{\pi_j\}$,

$$\pi_j = P(S_1 = s_j), \tag{7}$$

      which specifies the probability that the system starts in state s_j at time $t = 1$.

4. Model Training with Baum-Welch Algorithm

   a. Forward Algorithm: The training process of the Hidden Markov Model (HMM) using the Baum–Welch algorithm begins with the Forward Algorithm, which computes the probability of an observation sequence. This procedure consists of three main steps. In the initialization step,

$$\alpha_1(i) = \pi_i b_i(O_1), \tag{8}$$

   for $1 \leq i \leq N$, where N denotes the number of hidden states. In the induction step,

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i)\, a_{ij}\right] b_j(O_{t+1}), \tag{9}$$

   for $j = 1,2, \dots, N$ and $t = 1,2, \dots, T - 1$. The termination step computes the probability of the observation sequence given the model as

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i). \tag{10}$$

b. Backward Algorithm: It follows a similar principle but processes the observation sequence in reverse order. In the forward algorithm, the observation sequence is $\{O_1, O_2, \dots, O_t\}$, whereas in the backward algorithm, it becomes $\{O_{t+1}, O_{t+2}, \dots O_T\}$. The initialization step is

$$\beta_T(i) = 1, \tag{11}$$

for $i = 1, 2, \dots, N$. The induction step is

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j), \tag{12}$$

and the termination step is

$$P(O|\lambda) = \sum_{i=1}^{N} b_i(1)\pi(i)\beta_1(i). \tag{13}$$

c. Definition of two additional variables $\gamma_t(i)$ and $\xi_t(i,j)$. The first is $\gamma_t(i)$ representing the probability that the process is in state s_i at time t, given the observation sequence O and the model λ:

$$\gamma_t(i) = P(S_t = s_i|O,\lambda) = \frac{\alpha_t(i).\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i).\beta_t(i)}. \tag{14}$$

The second is $\xi_t(i,j)$, defined as the probability that the process is in state s_i at time t and transitions to state s_j at time $t+1$, given O and λ.

$$\begin{aligned} \xi_t(i,j) &= P\big(Q_t = s_i, Q_{t+1} = s_j \,\big|O,\lambda\big) \\ &= \frac{\alpha_t(i).\beta_{t+1}(j).b_j(O_{t+1}).a_{ij}}{P(O|\lambda)}. \end{aligned} \tag{15}$$

5. HMM Parameter Estimation

Once the state probabilities at time t, $\big(\gamma_t(i)\big)$, and the state transition probabilities at time t, $(\xi_t(i,j))$ have been computed, the parameters of the HMM $\big(\lambda = (A,B,\pi)\big)$ can be re-estimated using the following equations. The transition matrix is updated as

$$\widehat{a_{ij}} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)},$$

where the numerator represents the total number of transitions from s_i to state s_j and the denominator represents the total number of transitions originating from state s_i. The emission matrix is updated as

$$\widehat{b_j}(k) = \frac{\sum_{t=1}^{T} \gamma_t(j)}{\sum_{t=1}^{T-1} \gamma_t(i)}.$$

The estimate represents the ratio of the total probability of being in state s_j while producing observation v_k to the total probability of being in state s_j across all observation. Finally, the initial state probabilities are updated as

$$\widehat{\pi_j} = \gamma_1(i),$$

representing the probability that the system starts in state s_j at time $t = 1$.

6. Compute the next state distribution

Using the most recent state distribution obtained from the forward–backward or Viterbi algorithm γ_T, multiply by the state transition matrix A to obtain the probability distribution over hidden states at time $T+1$.

7. Predict the claim category and amount

   Multiply the predicted state distribution S_{T+1} by the emission matrix B to obtain the probability of each claim category. The category with the highest probability is selected as the predicted claim class, and, if required, it is converted to a nominal claim value using a representative statistic (e.g., historical mean or median for each category).

2.2 Convolutional Neural Networks - Long Short Term Memory (CNN-LSTM)

In addition to the Hidden Markov Model (HMM) approach, deep learning architectures are also explored for claim prediction to provide a comparative performance analysis. Among these, Long Short-Term Memory (LSTM) networks have been selected due to their proven effectiveness in modeling sequential data [16]. LSTM is a type of recurrent neural network (RNN) specifically designed to handle time-dependent sequences, capable of retaining both long-term and short-term information through its memory cell and gating mechanisms. By effectively mitigating the vanishing gradient problem that commonly occurs in conventional RNNs, LSTM is well-suited for modelling time series data such as insurance claims or daily/weekly financial trends [17], [18]. LSTMs have been extensively applied in domains including speech recognition, anomaly detection, and financial time series forecasting, often outperforming traditional statistical and conventional machine learning models [19].

Despite their effectiveness in capturing long-term dependencies, LSTM architectures face challenges in detecting short-term patterns or sudden fluctuations that may arise in datasets such as daily or weekly claim records. In these cases, LSTM tends to process each timestep independently without explicitly capturing the spatial correlation among neighboring features, which may contain valuable short-term patterns [20]. Therefore, to improve prediction accuracy and stability, a hybrid approach combining Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks (CNN–LSTM) is used [21]. CNNs have the ability to extract features and detect short-term patterns through convolutional operations, and when combined with LSTMs that capture temporal dependencies, the CNN–LSTM model can represent time series data more comprehensively and efficiently. In other words, the CNN acts as an initial feature extractor before the extracted information is passed to the LSTM for sequential processing, making this architecture particularly suitable for modelling complex and nonlinear time series data [16], [22].

The steps in claim prediction research using the CNN-LSTM model are as follows: [20]

1. Data preparation

   Historical insurance claim records were collected and aggregated by date to form a continuous time series. Prior to modelling, the dataset was normalized to the range $[0,1]$ using the MinMaxScaler transformation to ensure numerical stability during training. Sequential input windows were then constructed using a sliding-window approach with a fixed length of 90 timesteps, meaning that each input sequence of 90 consecutive days was used to predict the claim value on the subsequent day.

2. CNN-LSTM Model construction and compilation

   The proposed architecture consists of an initial 1D Convolutional (Conv1D) layer with 64 filters, a kernel size of 1, and a *tanh* activation function, followed by MaxPooling1D with a pool size of 1 to preserve local patterns. The extracted features are then fed into an LSTM layer with 64 units (*tanh* activation) to capture temporal dependencies, with a Dropout rate of 0.2 to mitigate overfitting. Finally, a Dense layer with a single neuron generates the output prediction. The model is compiled with the Adam optimizer and a learning rate of 0.001, and the loss function is the Mean Absolute Error (MAE), consistent with previous studies on time series forecasting [16]. The architecture is illustrated in Fig. 1.

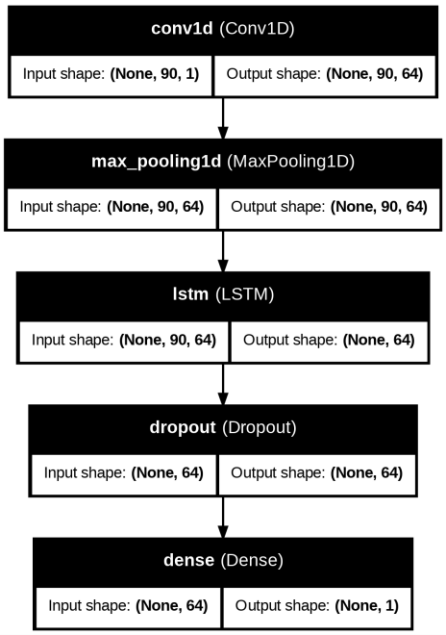

Figure 1. The Architecture of CNN-LSTM
Application Source: Google Colaboratory (Python-based environment).

3. Model training

   The CNN–LSTM model was trained on the prepared sequences using a mini-batch size of 64 for 50 epochs. A 10% validation split was used to monitor the model's generalization performance during training. This iterative process enabled the network to learn both short-term and long-term dependencies present in the historical claim data.

4. Prediction

   Once trained, the model was applied to the test dataset to produce forecasts of claim values. The predictions were subsequently transformed back to the original scale using the inverse MinMaxScaler transformation. Furthermore, the model was used to perform multi-step forecasting up to 5 days ahead, employing a recursive sliding-window strategy in which each predicted value was appended to the input sequence for the next prediction step.

5. Post-processing

   The predicted claim values were inverse-transformed to their nominal values in the original monetary unit. This step ensures that the outputs are directly interpretable in the context of real claim amounts, which is essential for practical decision-making.

3. RESULTS AND DISCUSSION

This section presents the results from two predictive models: the Hidden Markov Model (HMM) and the hybrid CNN–LSTM. The HMM predicted future claim categories and amounts based on probabilistic state transitions but largely replicated the discretized claim categories. In contrast, the CNN–LSTM, combining a CNN for short-term feature extraction and an LSTM for temporal modelling, produced more accurate and stable forecasts of continuous claim values over multiple steps ahead. The following subsections detail the dataset, preprocessing, and modelling results, beginning with a descriptive statistical analysis.

3.1 Descriptive Statistics

This study utilizes secondary data obtained from PT Asuransi Jasindo Syariah, consisting of historical general insurance claim records from the motor vehicle (KBM) line. The dataset covers the period from January 1, 2017, to November 5, 2021. A summary of the dataset's descriptive statistics is presented in Table 1. The dataset consists of 11,679 motor vehicle insurance claim records collected during the observation period.

Table 1. Descriptive Statistics of Data

Descriptive	Values
Count	11,679
Mean	IDR 9,566,487
Std	IDR 14,911,960
Min	IDR 41,250
Max	IDR 316,243,600

***Data source**: PT Asuransi Jasindo Syariah (official research data request)*

Following the descriptive summary, data visualization was conducted to provide a clearer understanding of the distribution of claim values. Based on the histogram distribution shown in Fig. 2, the claim severity data exhibit a strongly right-skewed distribution, where most claim amounts are concentrated in the lower range, while only a limited number of observations contain very large claim values. Visually, the majority of claims appear to be below IDR 20 million, whereas several extreme claims exceed IDR 100 million, with the maximum value reaching above IDR 300 million. This indicates that the dataset contains high-severity but low-frequency claim events, a common characteristic of insurance losses.

In addition, the boxplot demonstrates a large number of outliers located far above the upper quartile, suggesting substantial dispersion and heavy-tailed behavior in the claim distribution. The median claim value is close to the lower quartile, indicating that most policyholders submit relatively small claims, while a small proportion of claims accounts for a disproportionate share of the total loss exposure. Such characteristics imply that the data are non-normal, highly heterogeneous, and volatile over time, making traditional linear modelling approaches less suitable. Therefore, sequential and nonlinear models such as HMMs and LSTMs are appropriate for capturing latent claim patterns and temporal dependencies within the dataset.

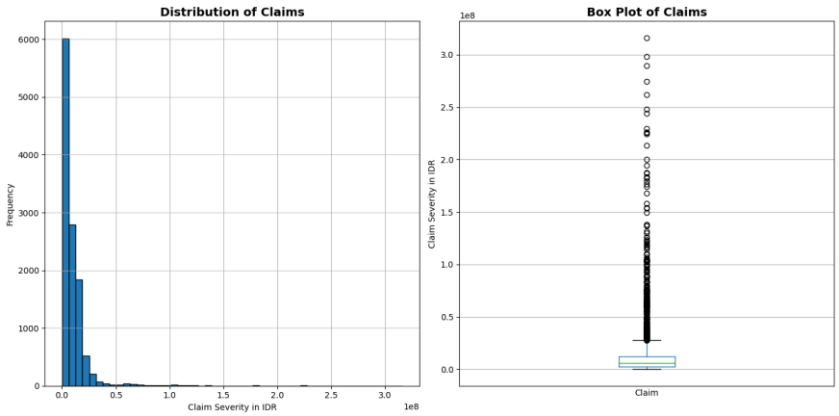

Figure 2. Histogram and Box Plot of Data
***Application Source**: Google Collaboratory (Python-based environment)*

3.2 Hidden Markov Model (HMM) Results

3.2.1 Claim Data Discretization

The claim values were discretized into three categories: low, medium, and high. This was achieved by applying quartile-based thresholds: 0–25% (low), 25–75% (medium), and 75–100% (high). Fig. 3 presents the frequency distribution of these categories, with Category 1 (medium claim) being the most prevalent, with approximately 5,900 occurrences, while Categories 0 (low claim) and 2 (high claim) each occur approximately 2,900 times. This indicates that the majority of claims in the dataset fall into the medium category.

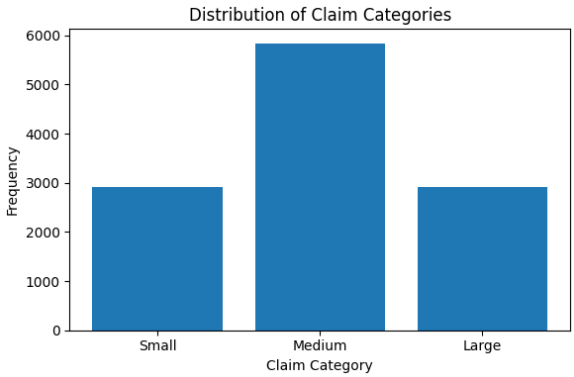

Figure 3. Bar Chart for Claim Category Distribution
***Application Source**: Google Collaboratory (Python-based environment)*

3.2.2. Parameters Estimation

The HMM parameters were estimated using the hmmlearn Python library. A maximum of 100 iterations was set to ensure model convergence, and a fixed random state of 42 was applied to guarantee reproducibility. The resulting model parameters include the transition and emission matrices (Figs. 4 (a) and (b)) and the predicted probability of each subsequent hidden state, as summarized in Table 2.

In this study, the hidden states were interpreted as latent insurance risk conditions underlying the claim behavior, namely normal risk (s_0), medium risk (s_1), and high risk (s_2). The transition matrix A represents the probability of transitioning from one latent risk condition to another between consecutive days. For example, the value $a_{11} = 0.99$ in Fig. 4 (a) indicates that when the system is in the medium-risk state, there is a 99% probability that the following day will remain in the same risk condition. Similarly, the value $a_{22} = 0.77$ shows that high-risk conditions also exhibit strong persistence over time, suggesting temporal clustering of large claim events in the dataset.

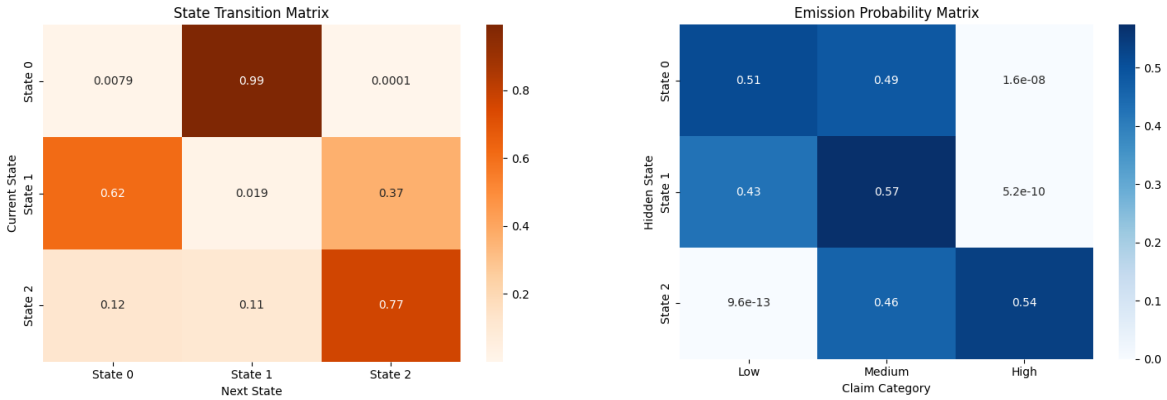

Figure 4. State Transition and Emission Probability Matrix
(a) Transition Matrix A, (b) Emission Matrix B
Application Source: Google Collaboratory (Python-based environment)

Table 2. Probability of The Next Hidden State

State	Probability	Average Claim (IDR)
s_0 (Normal Risk)	30.71%	4,069,521
s_1 (Medium Risk)	31.15%	4,512,150
s_2 (High Risk)	38.14%	18,121,784

***Data Source**: Author's experimental results.*

Meanwhile, the emission matrix B in Fig. 4 (b) represents the probability of observing a particular claim severity category given a specific hidden risk state. In this study, the observed variables correspond to the three categories of daily aggregate claim amounts (low, medium, and high). Fig. 4 (b) shows that hidden state (s_2) (high-risk state) has the highest probability of generating high-severity claims, whereas hidden state (s_0) is more frequently associated with low and medium claim categories. This result indicates that the hidden states reflect different underlying risk conditions within the insurance claim process. Accordingly, the

transition matrix explains how these risk conditions evolve over time, while the emission matrix describes how they are manifested in the observed daily claim severities.

A comparison between the predicted hidden states and the observed claim categories for the first 200 claims in Fig. 5 shows that the hidden states (red crosses) generally follow patterns similar to the observed claim categories (green circles). This suggests that the HMM successfully captures the dominant structure of claim severity in the dataset. However, the strong similarity between the hidden states and observed categories also indicates that the identified hidden patterns remain closely related to the predefined claim classifications, suggesting that the claim dynamics in the dataset may contain relatively limited hidden variability beyond the observed severity categories.

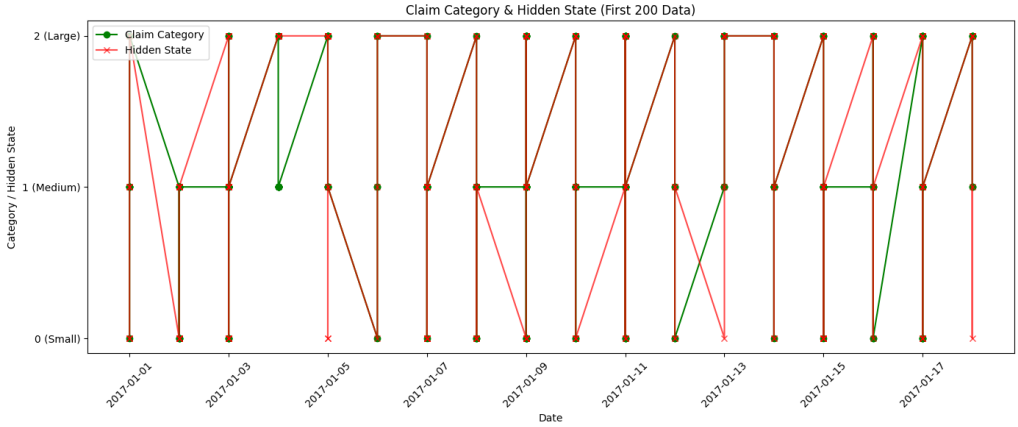

Figure 5. Comparison of Claim Categories and Predicted Hidden States for the First 200 Data
***Application Source**: Google Collaboratory (Python-based environment)*

3.3 CNN-LSTM Results

To develop the daily insurance claim prediction model, a hybrid deep learning architecture comprising a Convolutional Neural Network (CNN) and a Long Short-Term Memory (LSTM) was employed. This architecture is designed to effectively extract local temporal features through convolutional layers while capturing long-range dependencies using LSTM units. The model was trained in a supervised learning framework to perform regression on claim values, utilizing time-series input to reflect sequential patterns inherent in historical claim data.

Table 3. Parameter Setting of CNN-LSTM

Parameter Model	Value
Input Shape	(window_size=90, 1 fitur)
Conv1D Layer	filters=32, kernel_size=1, activation='tanh', padding='same'
MaxPooling1D	pool_size=1, padding='same'
LSTM Layer	units=64, activation='tanh'
Dropout Layer	dropout=0.2 (for regularization)
Dense (Output)	units=1 (claim value regression)
Optimizer	Adam
Learning Rate	0.001
Loss Function	mean_absolute_error
Epochs	50
Batch Size	64

***Data Source**: Author's parameter configuration for the proposed CNN-LSTM model in Google Colab (TensorFlow 2.15.0, Keras 2.15.0).*

Table 3 summarizes the detailed configuration of the CNN-LSTM model used in this study. The input structure uses a sliding window with a length of 90 time steps and a single feature per time step. The convolutional layer is configured with 32 filters, a kernel size of 1, and a hyperbolic tangent activation function, followed by a MaxPooling1D layer and an LSTM layer comprising 64 units with 'tanh' activation. To mitigate overfitting, a dropout rate of 0.2 is applied. The output layer is a dense unit for univariate

regression. Model optimization is performed using the Adam optimizer with a learning rate of 0.001 and Mean Absolute Error (MAE) as the loss function, trained over 50 epochs with a batch size of 64.

The model architecture is summarized in Table 4, detailing each computational layer and its corresponding parameter count. The convolutional layer extracts 64 feature maps from the input sequence, followed by a pooling layer that preserves the temporal resolution. These features are then passed to an LSTM layer with 64 units, which captures sequential dependencies in the data. A dropout layer is incorporated to mitigate overfitting, and the final dense layer performs the regression output. Overall, the model comprises 99,653 parameters, of which 33,217 are trainable, highlighting the balance between model complexity and learning capacity suited for time-series prediction tasks.

Table 4. Model Architecture Summary of the Proposed CNN-LSTM Network

Layer (type)	Output Shape	Parameter
Conv1d	(none, 90, 64)	128
Max_pooling1d	(none, 90, 64)	0
Lstm	(none, 64)	33,024
Droput	(none, 64)	0
Dense	(none, 1)	65
Total parameters: 99.653 (389,27 KB)		
Trainable parameters: 33.217 (129,75 KB)		
Non-trainable parameters 0 (0,00 B)		
Optimizer Parameters: 66.436 (259,52 KB)		

***Data Source**: Model summary output generated from experiments in Google Collab using TensorFlow.*

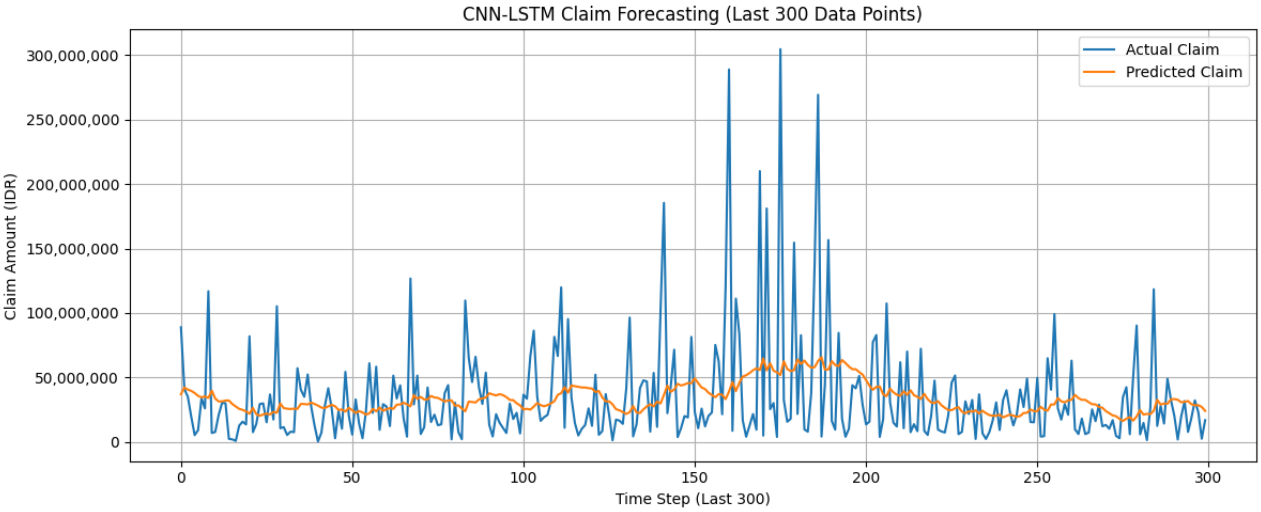

Figure 6. Forecasting of Daily Insurance Claims Using CNN-LSTM (Last 300 Observations)
Application Source: Google Collaboratory (Python-based environment)

Fig. 6 illustrates the forecasting performance of the CNN-LSTM model on the last 300 data points of daily insurance claims. The blue line represents the actual claim values, which display considerable fluctuations and several extreme peaks, indicating the presence of significant outliers or unusual claim events. Meanwhile, the orange line shows the predicted values, which generally capture the underlying trend and seasonality of the actual data, albeit with limited responsiveness to sharp spikes. This result suggests that the CNN-LSTM model, while effective at 2721 temporal patterns and smoothing out noise, still requires enhancement in sensitivity toward abrupt changes.

To improve the model's predictive accuracy and adaptability in the presence of volatile patterns and irregularities, this study adopts an ensemble learning approach that integrates Long Short-Term Memory (LSTM) networks with an attention mechanism and the Extreme Gradient Boosting (XGBoost) algorithm. The attention mechanism enables the LSTM component to assign dynamic weights to different time steps, thereby emphasizing critical temporal features that contribute significantly to prediction outcomes. In parallel, XGBoost captures complex nonlinear relationships and exhibits strong resilience to outliers, making the ensemble architecture well-suited for high-variability time-series forecasting tasks.

The effectiveness of such a hybrid approach has been demonstrated in recent empirical studies. Din et al. [23] introduced the ACB-XDE framework, combining attention-based BiLSTM and XGBoost, and reported substantial improvements in predictive accuracy, with reductions in MAE and RMSE of 53% and 38%, respectively, compared to single-model baselines. Similarly, Rahmani et al. [24] proposed a hybrid model incorporating attention-driven CNN-LSTM with XGBoost for short-term electricity price forecasting and observed significant error reduction across multiple evaluation metrics. These findings underscore the potential of Attention-LSTM + XGBoost ensembles to enhance predictive performance in time-series applications involving noisy, highly dynamic data. Table 5 summarizes the detailed parameters of the Ensemble model used in this study.

Table 5. Parameter Setting of Model Ensemble

Parameters	Value
Input feature dimension	1
Input time step (lookback)	90
LSTM hidden units	64
LSTM layer type	Attention-LSTM
LSTM activation function	default (tanh assumed)
Loss function (LSTM)	Mean Squared Error (MSE)
Optimizer (LSTM)	Adam
Learning rate (LSTM)	0.001
Batch size	32
Epochs (LSTM training)	20
XGBoost n_estimators	100
XGBoost learning_rate	0.1
XGBoost max_depth	4
XGBoost objective	reg:squarederror (default)
Target of XGBoost	Residual from LSTM
Final prediction	LSTM output + XGBoost residual

***Data Source**: Author's parameter configuration for the proposed ensemble model in Google Colab (TensorFlow 2.15.0, Keras 2.15.0)*

Fig. 7 presents the forecasting results using the Ensemble model (LSTM Attention + XGBoost), showing that the predicted values closely follow the actual claim trends, including short-term fluctuations and peak values. The integration of LSTM Attention effectively captures long-term temporal dependencies, while XGBoost enhances the model's ability to fit non-linear patterns, resulting in more accurate predictions that align with actual observations.

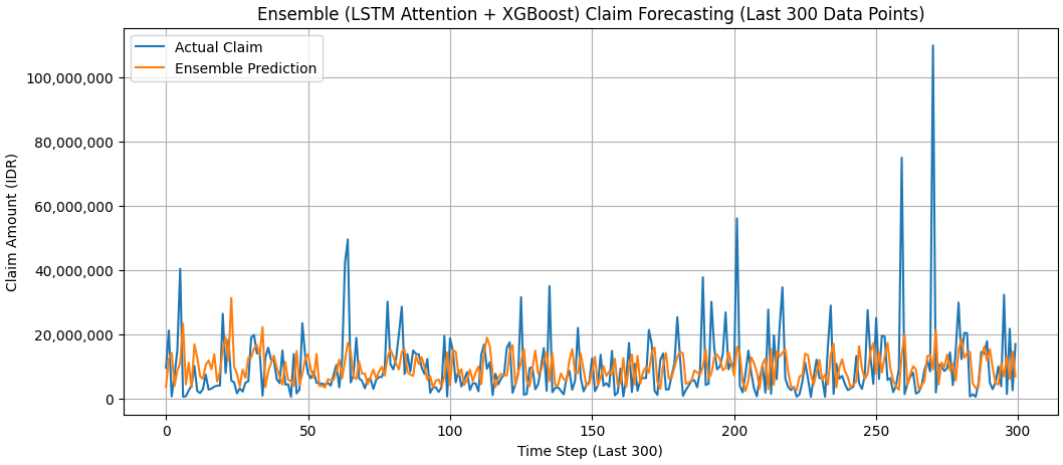

Figure 7. Forecasting of Daily Insurance Claims Using Ensemble (Last 300 Observations)
Application Source: Google Collaboratory (Python-based environment)

Table 6. Evaluation Each Model

Model	MAE	RMSE	R-squared
HMM (Algorithm Baum-Welch)	4,380,650	12,356,600	0.31
CNN LSTM	26,778,684	40,257,041	0.0269
Ensemble (Attention-LSTM + XGBoost)	7,070,704	14,616,430	0.0241

***Data Source**: Author's experimental results.*

Based on the model evaluation results on the testing dataset presented in Table 6, the HMM statistical model using the Baum–Welch algorithm achieved the best predictive performance for daily motor vehicle insurance claim prediction, with an MAE of 4,380,650, RMSE of 12,356,600, and (R^2) of 0.31. The evaluation was conducted using the same training–testing data partition for all models, where the testing set consisted of previously unseen daily claim observations to ensure an objective and reproducible comparison. The HMM model outperformed the CNN-LSTM model (MAE: 26,778,684; RMSE: 40,257,041; (R^2): 0.0269) as well as the Attention-LSTM + XGBoost ensemble model (MAE: 7,070,704; RMSE: 14,616,430; (R^2): 0.0241). These findings indicate that the probabilistic structure of HMM is more suitable for modelling the daily aggregated motor vehicle claim data used in this study.

The descriptive analysis shows that the claim data exhibit highly skewed distributions, characterized by frequent low-severity claims and occasional extreme claim spikes. Such patterns suggest alternating claim risk conditions over time, including transitions between relatively stable periods with low claim activity and periods with unusually high claim amounts. HMM is particularly effective in capturing these temporal transitions because it represents the claim process as a sequence of hidden states underlying the observed claim severities. Through the transition probabilities between hidden states, the model can characterize changes in risk conditions across consecutive days, which aligns well with the stochastic nature of motor vehicle insurance claims [6], [17].

In contrast, the CNN-LSTM and Attention-LSTM + XGBoost ensemble models showed relatively lower predictive performance. This result is likely influenced by the characteristics of the dataset, which consists of 11,679 daily aggregated claim observations with moderate temporal complexity and substantial volatility. Although deep learning models such as LSTMs are theoretically capable of learning nonlinear temporal relationships, they generally require larger datasets and more complex sequential patterns to achieve strong generalization performance [10]. With limited data variability and a moderate dataset size, these models are more susceptible to overfitting, leading them to learn short-term fluctuations and noise rather than the underlying claim dynamics.

Compared with deep learning approaches, the HMM framework is more parameter-efficient and relies on probabilistic state-transition assumptions that closely match the observed structure of the claim process [25]. The daily motor vehicle claim series appears to contain relatively limited hidden complexity beyond the observed severity categories, making the simpler HMM framework more stable and interpretable for this prediction task. Therefore, these findings suggest that, for daily-aggregated motor vehicle insurance claim prediction with moderate-scale data and fluctuating claim severity patterns, classical probabilistic approaches such as HMMs may still provide more robust and reliable performance than more complex deep learning architectures.

4. CONCLUSION

This study compared the predictive performance of the Hidden Markov Model (HMM) and CNN-LSTM in modeling temporal patterns and predicting daily motor vehicle insurance claims. The evaluation results showed that the HMM model achieved better predictive performance than the CNN-LSTM model, as indicated by lower MAE and RMSE values and a higher. These findings suggest that the probabilistic state-transition mechanism of an HMM is more suitable for capturing the fluctuating, highly skewed characteristics of daily motor vehicle insurance claim data, which are dominated by low-severity claims with occasional extreme claim spikes.

In addition, the Attention-LSTM + XGBoost ensemble model was included as a supplementary deep learning benchmark to evaluate whether more advanced hybrid architectures could improve prediction performance. Although the ensemble model performed better than the standalone CNN-LSTM model, it still did not outperform HMM. Therefore, the study concludes that classical probabilistic models such as HMM remain effective and competitive for daily motor vehicle insurance claim prediction, particularly for moderate-scale datasets with relatively limited temporal complexity.

Author Contributions

Ainun Mawaddah Abdal: Conceptualization, Methodology, Supervision, Writing - Original Draft. Andi Muhammad Anwar: Data curation, Investigation, Validation. Illuminata Wynnie: Formal Analysis, Writing

- Review and Editing. Amil Siddik: Software, Methodology, Resources. Edy Saputra Rusdi: Visualization, Data Curation. Mauliddin: Project administration, Resources. All authors discussed the results and contributed to the final manuscript.

Funding Statement

This research was funded by the Internal Research Grant of the Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Hasanuddin (Grant No. 17251/UN4.11/HK.04/2024).

Acknowledgment

The authors gratefully acknowledge the financial support provided by the Internal Research Grant of the Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Hasanuddin, which enabled the successful completion of this study. The authors also thank PT Asuransi Jasindo Syariah for granting access to the insurance claim dataset used in this study.

Declarations

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Declaration of Generative AI and AI-assisted technologies

The authors used generative AI (ChatGPT) only to assist with language polishing and formatting consistency (e.g., improving wording and ensuring uniform terminology). No AI was used to generate research content, perform analyses, or create/modify figures and tables. The authors reviewed the manuscript in full and remain responsible for its content.

REFERENCES

[1] A. Bücher and A. Rosenstock, "MICRO-LEVEL PREDICTION OF OUTSTANDING CLAIM COUNTS BASED ON NOVEL MIXTURE MODELS AND NEURAL NETWORKS," *Eur. Actuar. J.*, vol. 13, no. 1, pp. 55–90, Jun. 2023, doi: https://doi.org/10.1007/s13385-022-00314-4.

[2] B. Avanzi, Y. Li, B. Wong, and A. Xian, "ENSEMBLE DISTRIBUTIONAL FORECASTING FOR INSURANCE LOSS RESERVING," *Scand. Actuar. J.*, vol. 2024, no. 9, pp. 971–1012, 2024, doi: https://doi.org/10.1080/03461238.2024.2365392.

[3] Verawaty Ompusunggu, "5 INSURANCE COMPANIES THAT WENT BANKRUPT AND DEFAULTED IN INDONESIA," Lifepal. Accessed: May 10, 2025. [Online]. Available: https://lifepal.co.id/media/perusahaan-asuransi/perusahaan-asuransi-yang-bangkrut/

[4] H. A. Surya, Sukono, H. Napitupulu, and N. Ismail, "A SYSTEMATIC LITERATURE REVIEW OF INSURANCE CLAIMS RISK MEASUREMENT USING THE HIDDEN MARKOV MODEL," *Risks*, vol. 12, no. 11, p. 169, Oct. 2024, doi: https://doi.org/10.3390/risks12110169

[5] M. T. Jatipaningrum, K. Suryowati, Libertania, M. Melania, and E. Un, "PREDICTING THE RUPIAH EXCHANGE RATE AGAINST THE DOLLAR USING FTS-MARKOV CHAINS AND HIDDEN MARKOV MODELS," *Jurnal Derivat*, vol. 6, no. 1, pp. 32–41, Jul. 2019, doi: https://doi.org/10.31316/j.derivat.v6i1.334.

[6] A. Alimansyah and A. Purqon, "APPLICATION OF HIDDEN MARKOV MODEL IN STOCK PRICE PREDICTION IN INDONESIA," *Bahasa*, pp. 102–08, Jul. 2016, Accessed: Mar. 01, 2025. [Online]. Available: https://ifory.id/proceedings/2016/4chQ7E9Cp/snips_2016_arfian_alimansyah_1b09d74502f3144edef531cc4bda41db.pdf

[7] M. A. Istiake Sunny, M. M. S. Maswood, and A. G. Alharbi, "DEEP LEARNING-BASED STOCK PRICE PREDICTION USING LSTM AND BI-DIRECTIONAL LSTM MODEL," in *2nd Novel Intelligent and Leading Emerging Sciences Conference, NILES 2020*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020, pp. 87–92. doi: https://doi.org/10.1109/NILES50944.2020.9257950.

[8] I. Cohen Sabban, O. Lopez, and Y. Mercuzot, "AUTOMATIC ANALYSIS OF INSURANCE REPORTS THROUGH DEEP NEURAL NETWORKS TO IDENTIFY SEVERE CLAIMS," *Annals of Actuarial Science*, vol. 16, no. 1, pp. 42–67, Mar. 2022, doi: https://doi.org/10.1017/S174849952100004X.

[9] L. Alzubaidi *et al.*, "REVIEW OF DEEP LEARNING: CONCEPTS, CNN ARCHITECTURES, CHALLENGES, APPLICATIONS, FUTURE DIRECTIONS," *J. Big Data*, vol. 8, no. 1, Dec. 2021, doi: https://doi.org/10.1186/s40537-021-00444-8.

[10] L. Zhang and D. Jánošík, "ENHANCED SHORT-TERM LOAD FORECASTING WITH HYBRID MACHINE LEARNING MODELS: CATBOOST AND XGBOOST APPROACHES," *Expert Syst. Appl.*, vol. 241, May 2024, doi: https://doi.org/10.1016/j.eswa.2023.122686.

[11] D. C. R Novitasari *et al.*, "WIND SPEED ANALYSIS IN TIDAL WATER USING THE FORWARD-BACKWARD ALGORITHM IN HIDDEN MARKOV MODELS IN THE TANJUNG PERAK PORT AREA OF SURABAYA," *Jurnal Sains Matematika dan Statistika*, vol. 4, no. 1, 2018, doi: http://dx.doi.org/10.24014/jsms.v4i1.5254 .

[12] C. Wang, K. Li, and X. He, "NETWORK RISK ASSESSMENT BASED ON BAUM WELCH ALGORITHM AND HMM," *Mobile Networks and Applications*, vol. 26, no. 4, pp. 1630–1637, Aug. 2021, doi: https://doi.org/10.1007/s11036-019-01500-7.

[13] G. F. Dar, T. R. Padi, S. Rekha, and Q. F. Dar, "STOCHASTIC MODELING FOR THE ANALYSIS AND FORECASTING OF STOCK MARKET TREND USING HIDDEN MARKOV MODEL," *Asian Journal of Probability and Statistics*, pp. 43–56, Jun. 2022, doi: https://doi.org/10.9734/ajpas/2022/v18i130436.

[14] Z. Su and B. Yi, "RESEARCH ON HMM-BASED EFFICIENT STOCK PRICE PREDICTION," *Mobile Information Systems*, vol. 2022, pp. 1–8, Mar. 2022, doi: https://doi.org/10.1155/2022/8124149.

[15] I. Sassi, S. Anter, and A. Bekkhoucha, "A NEW IMPROVED BAUM-WELCH ALGORITHM FOR UNSUPERVISED LEARNING FOR CONTINUOUS-TIME HMM USING SPARK," *International Journal of Intelligent Engineering and Systems*, vol. 13, no. 1, pp. 214–226, Feb. 2020, doi: https://doi.org/10.22266/ijies2020.0229.20.

[16] S. Kim and M. Kang, "FINANCIAL SERIES PREDICTION USING ATTENTION LSTM," Feb. 2019, doi: https://doi.org/10.48550/arXiv.1902.10877 .

[17] S. Trihandaru, H. A. Parhusip, A. W. Goni, S. Trihandaru, H. A. Parhusip, and A. W. Goni, "OVERCOMING OVERFITTING IN MONKEY VOCALIZATION CLASSIFICATION: USING LSTM AND LOGISTIC REGRESSION," *BAREKENG: J. Math. & App*, vol. 19, no. 2, pp. 973–0986, 2025, doi: https://doi.org/10.30598/barekengvol19iss2pp973-986

[18] M. A. Istiake Sunny, M. M. S. Maswood, and A. G. Alharbi, "DEEP LEARNING-BASED STOCK PRICE PREDICTION USING LSTM AND BI-DIRECTIONAL LSTM MODEL," in *2nd Novel Intelligent and Leading Emerging Sciences Conference, NILES 2020*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020, pp. 87–92. doi: https://doi.org/10.1109/NILES50944.2020.9257950.

[19] C. Yu *et al.*, "GRADIENT BOOSTING DECISION TREE WITH LSTM FOR INVESTMENT PREDICTION," May 2025, doi: https://doi.org/10.1109/ACCTCS66275.2025.00017.

[20] W. Lu, J. Li, Y. Li, A. Sun, and J. Wang, "A CNN-LSTM-BASED MODEL TO FORECAST STOCK PRICES," *Complexity*, vol. 2020, no. 1, Nov. 2020, doi: https://doi.org/10.1155/2020/6622927

[21] W. Gamaleldin, O. Attayyib, M. M. Alnfiai, F. A. Alotaibi, and R. Ming, "A HYBRID MODEL BASED ON CNN-LSTM FOR ASSESSING THE RISK OF INCREASING CLAIMS IN INSURANCE COMPANIES," *PeerJ Comput. Sci.*, vol. 11, 2025, doi: https://doi.org/10.7717/peerj-cs.2830/supp-1

[22] S. Mehtab and J. Sen, "STOCK PRICE PREDICTION USING CNN AND LSTM-BASED DEEP LEARNING MODELS," in *2020 International Conference on Decision Aid Sciences and Application, DASA 2020*, Institute of Electrical and Electronics Engineers Inc., Nov. 2020, pp. 447–453. doi: https://doi.org/10.1109/DASA51403.2020.9317207

[23] R. U. Din, S. Ahmed, S. H. Khan, A. Albanyan, J. Hoxha, and B. Alkhamees, "A NOVEL DECISION ENSEMBLE FRAMEWORK: ATTENTION-CUSTOMIZED BILSTM AND XGBOOST FOR SPECULATIVE STOCK PRICE FORECASTING," *PLoS One*, vol. 20, no. 4 April, Apr. 2025, doi: https://doi.org/10.1371/journal.pone.0320089

[24] A.-H. Rahmani, J. Salehi, and M. Shafie-Khah, "AN ADVANCED HYBRID CNN-LSTM-XGBOOST MODEL WITH WAVELET TRANSFORM AND ATTENTION MECHANISM FOR ACCURATE SHORT-TERM ELECTRICITY PRICE FORECASTING," Jun. 2025. doi: https://dx.doi.org/10.2139/ssrn.5314801

[25] M. S. Hossain and F. Parvin, "A COMPARATIVE STUDY OF VARIOUS STATISTICAL AND MACHINE LEARNING MODELS FOR PREDICTING RESTAURANT DEMAND IN BANGLADESH," *PLoS One*, vol. 20, no. 6 June, Jun. 2025, doi: https://doi.org/10.1371/journal.pone.0325449