Central European Journal of Economic Modelling and Econometrics

The Long Short-Term Memory Algorithm and the Autoregressive Integrated Moving Average Approach in Business Tendency Survey

Ewa Ratuszny*

Submitted: 12.09.2024, Accepted: 27.05.2025

Abstract

This study investigates the effectiveness of machine learning models in forecasting construction indicators derived from Business Tendency Survey data. Specifically, we compare the performance of traditional statistical models such as the autoregressive integrated moving average (ARIMA) with long short-term memory (LSTM) networks and hybrid approaches combining both. Using a range of economic variables – including sector and economic evaluations, production, financial situation, investments, and sentiment indicator (IRGBUD) – we evaluate model accuracy across testing dataset and rolling forecast strategy to assess consistency over time. Results demonstrate that while LSTM networks capture non-linear dependencies and temporal patterns, ARIMA-based models consistently outperforms LSTM in scenarios involving seasonal and cyclical structures. The findings highlight that the choice of model should align with the nature of the time series, particularly in relation to seasonality, volatility, and trend dynamics. This work offers practical implications for improving economic forecasting with machine learning in survey-based environments.

Keywords: ARIMA, LSTM, business tendency survey, neural network

JEL Classification: C22, C53, C45, L74, E32

*Research Institute of Economic Development, SGH Warsaw School of Economics, Poland; e-mail: eratus@sgh.waw.pl; ORCID: 0000-0002-3187-8786

1 Introduction

There are various machine learning tools and techniques that can be effectively applied to analyze business tendency survey data. These include regression analysis, decision trees, ensemble methods such as random forests, support vector machines (SVM), neural networks, and natural language processing (NLP). Each of these approaches provides unique strengths – from identifying statistical relationships and classification patterns to uncovering complex, nonlinear dynamics in large and multidimensional datasets.

Such tools enable businesses to extract deeper insights from survey data and support more accurate forecasting and decision-making. In this research, we focus on neural networks, which are among the most advanced and powerful frameworks in contemporary machine learning. Numerous studies have explored the application of neural networks to economic forecasting and business surveys. For instance, Zhang (2004) highlights their use across various domains such as market response, stock returns, emerging market indices, tourism demand, and consumer behavior analysis. Similarly, Zhong and Enke (2017) presents hybrid machine learning approaches combining neural networks with traditional models to forecast economic indicators. Their work introduces a comprehensive big data analytics process applied to stock market return prediction.

Recent contributions also emphasize the growing relevance of combining neural networks with traditional statistical models in forecasting tasks. For example, Michańków et al. (2024) examine the hedging properties of algorithmic investment strategies using LSTM and ARIMA-GARCH models for equity indices. Likewise, Kashif and Ślepaczuk (2024) propose a hybrid LSTM-ARIMA approach in algorithmic investment strategies, showing the advantages of model integration. Additional research by Roszyk and Ślepaczuk (2022) explores volatility forecasting of the S&P 500 using a hybrid approach that ensembles VIX, GARCH, and LSTM models. Moreover, Michańków et al. (2022) provide a comparative analysis of LSTM applications in investment strategies involving BTC and S&P 500 indices, while Baranochnikov and Ślepaczuk (2023) evaluate the architecture of LSTM and Gated Recurrent Unit (GRU) models using a novel walk-forward validation method in the context of algorithmic investment strategy.

The contributions such as Hamid and Iqbal (2004), Moshiri et al. (1999), and Taylor and Buizza (2003) also provide evidence of the effectiveness of neural networks in economic and financial applications. These studies collectively underscore the growing importance of machine learning methods, particularly neural networks and hybrid models, in enhancing the accuracy of forecasts derived from business tendency surveys. The primary objective of this research is to evaluate and compare the forecasting performance of traditional statistical models and modern machine learning techniques in the context of macroeconomic indicators derived from business tendency surveys. Specifically, the study investigates the effectiveness of LSTM networks – both univariate and multivariate – against the classical ARIMA model and a hybrid ARIMA+LSTM approach. By conducting extensive model training, validation, and sensitivity analysis, the research aims to identify which model architecture provides the most accurate and robust predictions across different forecasting horizons. The findings are intended to guide the application of machine learning methods in economic forecasting and support more informed decision-making for policy-makers and analysts.

2 Forecasting methodology

This study investigates the predictive performance of classical and deep learning models in the context of short-term economic forecasting within Poland's construction sector. The models explored include the traditional ARIMA model, univariate and multivariate LSTM networks, and a hybrid ARIMA-LSTM approach that combines the strengths of both techniques.

2.1 Autoregressive integrated moving average model

The autoregressive integrated moving average model, denoted as ARIMA(p,d,q), is a widely used statistical approach for time series forecasting. The parameters p, d, and q represent the order of the autoregressive component, the degree of differencing, and the order of the moving average component, respectively. An ARIMA model is particularly suited for modeling univariate time series data that exhibit non-stationarity, which can be removed through differencing.
The ARIMA model extends the autoregressive moving average (ARMA) process by incorporating differencing of the series. The general form of an ARMA(p, q) process is defined as follows (Tsay, 2005):

$$X_t - \alpha 1 X t - 1 - \cdots - \alpha p X t - p = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} \tag{1}$$

where α_i are the autoregressive coefficients, θ_i are the moving average coefficients, and ε_t represents white noise error terms, typically assumed to be independently and identically distributed (i.i.d.) with a normal distribution and zero mean.
While the standard ARIMA model captures linear trends and short-term autocorrelation, it is often necessary to account for regular seasonal fluctuations in economic time series data. In such cases, the seasonal ARIMA (SARIMA) model is employed. SARIMA extends ARIMA by including seasonal autoregressive and moving average terms, along with seasonal differencing. The SARIMA model is generally expressed as ARIMA(p, d, q) $\times$ (P, D, Q)$_s$, where P, D, and Q denote the seasonal components and s is the length of the seasonal cycle.
Model identification, parameter estimation, and diagnostic checking in ARIMA and SARIMA frameworks are thoroughly discussed in the literature. Box et al. (2015) provides a comprehensive treatment of these processes, including strategies

for dealing with model selection and residual analysis. Furthermore, Hyndman and Athanasopoulos (2018) offers a practical and accessible approach to time series modeling and forecasting, with particular emphasis on using SARIMA models to effectively capture recurring seasonal patterns in macroeconomic and financial data. In this study, SARIMA models are employed to generate forecasts for quarterly indicators in the construction sector. The choice of model parameters is guided by statistical tests for stationarity, seasonal decomposition analysis, and automated model selection procedures. The results obtained from SARIMA models are later compared to those produced by LSTM and hybrid modeling approaches to evaluate forecasting performance.

2.2 Long short-term memory approach

Neural networks offer a powerful framework for analyzing business tendency survey data due to their ability to detect complex, nonlinear relationships and extract meaningful insights from large, multivariate datasets. As demonstrated by Zhang (2004), neural networks can be effectively applied to tasks such as pattern recognition, forecasting, feature extraction, sentiment analysis, and predictive modeling, making them well-suited for economic applications.

Compared to traditional statistical approaches such as ARIMA, neural networks – particularly recurrent architectures like LSTM – provide advantages in handling nonlinear dynamics and uncovering hidden patterns in qualitative and semi-quantitative data such as business surveys. The LSTM network was introduced by Sepp Hochreiter and Jürgen Schmidhuber in 1997 as a solution to the vanishing gradient problem encountered in traditional recurrent neural networks (RNNs) (Hochreiter and Schmidhuber, 1997). LSTM networks have been widely applied to time series forecasting tasks, but in this study, they are used specifically to model the evolution of business sentiment indicators drawn from survey data in the construction sector. The models are trained to capture overall dynamics reflected in respondents' assessments of the economic situation, business conditions, and future expectations.

LSTM networks are particularly effective when the underlying data exhibit long-term dependencies or delayed effects, which is often the case with business sentiment indicators influenced by macroeconomic and sectoral developments. The architecture is well-suited for this task because it can learn from sequences of historical survey responses and generate forecasts that integrate both recent and longer-term patterns in respondent behavior.

The LSTM network is a specialized form of RNN designed to retain memory over long sequences. It achieves this through the introduction of memory cells and gating mechanisms. Each LSTM unit receives three main inputs: the input vector at the current time step (X_t), the hidden state from the previous step (h_{t-1}), and the cell state or memory from the previous unit (C_{t-1}). The outputs of the current unit are the updated hidden state (h_t) and the new cell state (C_t), which carry forward the short-term and long-term memory, respectively (Aggarwal, 2024).

The internal mechanism of an LSTM unit is governed by three key gates: the forget gate, the input gate, and the output gate. These gates regulate the retention, update, and propagation of information. Figure 1 presents the LSTM architecture diagram, illustrating the internal structure of a single memory cell, including its input, forget, and output gates.
The forget gate determines the proportion of the previous cell state to retain, calculated as:

$$f_t = \sigma(W_f \cdot [h_{t-1}, X_t] + b_f) \tag{2}$$

The input gate assesses how much of the new information should be stored in the cell state:

$$i_t = \sigma(W_i \cdot [h_{t-1}, X_t] + b_i) \tag{3}$$

The candidate values for updating the memory are generated through a tanh activation:

$$\hat{C}_t = \tanh(W_c \cdot [h_{t-1}, X_t] + b_c) \tag{4}$$

These candidate values are combined with the input gate output and the previous memory state to form the updated cell state:

$$C_t = i_t \cdot \hat{C}_t + f_t \cdot C_{t-1} \tag{5}$$

The output gate determines which parts of the updated cell state contribute to the output hidden state. It is calculated as:

$$o_t = \sigma(W_o \cdot [h_{t-1}, X_t] + b_o) \tag{6}$$

Finally, the new hidden state is produced by modulating the cell state using the output gate and a tanh transformation:

$$h_t = o_t \cdot \tanh(C_t) \tag{7}$$

The LSTM cell utilizes weight matrices and biases associated with each gate:

i) W_f, W_i, W_o, W_c are weight matrices for the forget, input, output, and cell candidate gates, respectively.

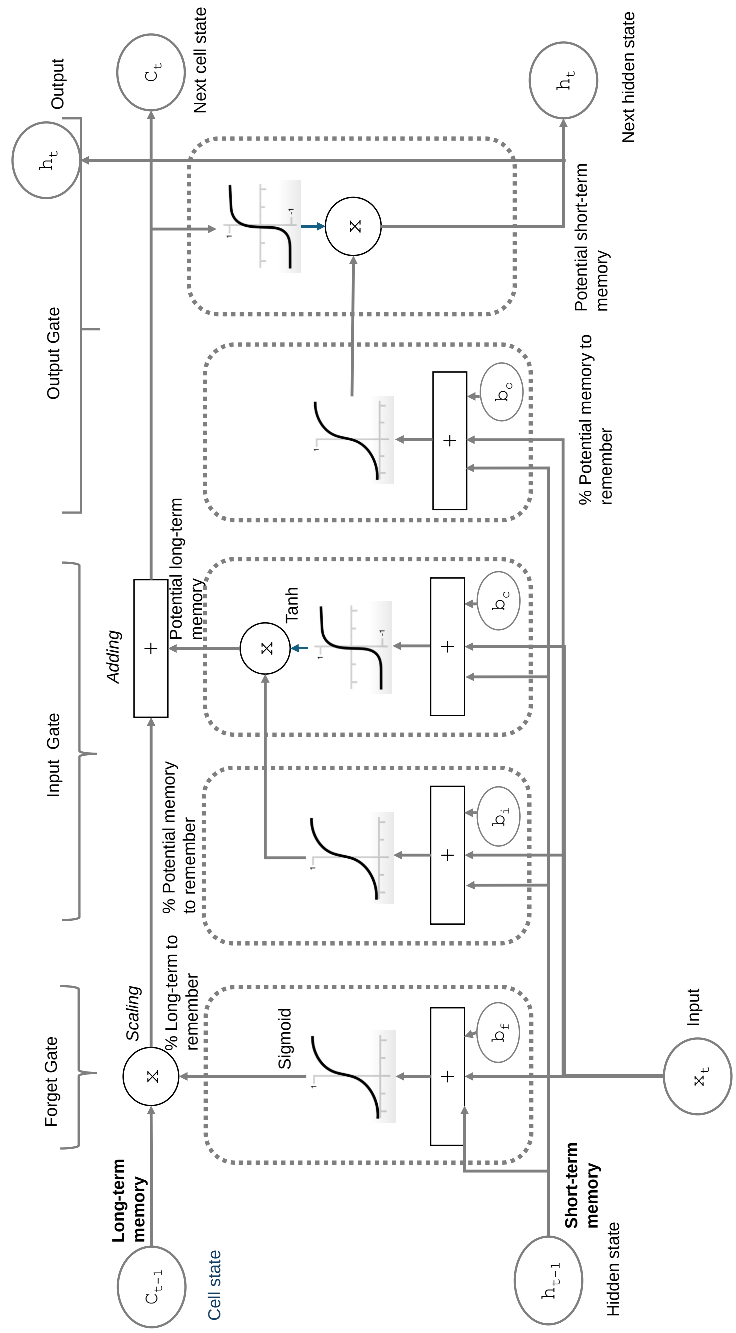
ii) b_f, b_i, b_o, b_c are corresponding bias vectors.

iii) f_t (forget gate), i_t (input gate), and o_t (output gate) are vectors of values in $[0, 1]$ that control which information is retained, updated, or output.

These parameters are learned through backpropagation during training. Sigmoid activation functions, defined as:

$$\sigma(x) = \frac{1}{1 + e^{-x}}, \tag{8}$$

Figure 1: Diagram of Long Short-Term Memory



Source: Diagram reproduced from Yan (2024).

are used within the gates to produce outputs in the range of $[0, 1]$, representing the proportion of information retained or discarded.
In contrast, the tanh activation function,

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \tag{9}$$

is employed to regulate the values of memory updates, allowing both positive and negative contributions to the cell state.
The architecture of the LSTM network allows for separate handling of long-term and short-term dependencies. The cell state C_t flows through the network relatively unimpeded, ensuring stable memory over long periods. This mechanism mitigates the gradient decay seen in standard RNNs and enhances the model's ability to learn time-dependent patterns, including business cycles and seasonal dynamics. As noted by GeeksforGeeks (2024), the dual-path structure of LSTM – with separate memory flows for long-term and short-term components – enables accurate and interpretable sequence modeling.

2.3 Hybrid ARIMA and LSTM modeling framework

To enhance forecasting performance and capture both linear and nonlinear dynamics present in business tendency survey data, this study implements a hybrid modeling strategy that combines the Autoregressive Integrated Moving Average model with a Long Short-Term Memory network. This hybrid approach has been widely recognized in the literature as an effective method for handling complex, nonstationary, and nonlinear time series data (Zhang et al., 2003; Ahmed et al., 2010; Moshiri et al., 1999).
The hybrid model leverages the complementary strengths of ARIMA and LSTM. ARIMA, as a classical statistical method, is proficient in modeling linear structures, autocorrelation, and seasonality within time series data. Its interpretability and well-established theoretical underpinnings make it suitable for extracting trend and cyclical components (Box et al., 2015; Hyndman and Athanasopoulos, 2018). However, ARIMA is limited in its ability to model nonlinearities and long-range dependencies. Conversely, LSTM networks – a variant of Recurrent Neural Networks (RNNs) – are designed to capture temporal patterns, nonlinear relationships, and long-term dependencies via gated memory mechanisms (Hochreiter and Schmidhuber, 1997; Bengio, 2012).
The construction of the hybrid model involves two sequential stages:
In the first stage, an ARIMA or SARIMA model is fitted to the original univariate time series. Model order selection is based on the Bayesian Information Criterion (BIC), while the augmented Dickey-Fuller (ADF) test is used to determine the need for differencing and to assess stationarity (Tsay, 2005). When seasonal effects are detected, a seasonal ARIMA (SARIMA) specification is used. Once fitted, ARIMA in-

sample forecasts are generated, and residuals are computed as the difference between the actual and fitted values.
In the second stage, these residuals are treated as a new time series and modeled using an LSTM network. The rationale is that the residuals contain nonlinear and unexplained patterns that ARIMA could not capture. Training the LSTM on the residual sequence helps isolate and learn those nonlinear characteristics without being constrained by the linear assumptions of ARIMA (Zhang et al., 2003). The residuals are framed into supervised learning format using a sliding window approach, and the LSTM model is optimized via grid search across key hyperparameters, including sequence length, number of layers, neurons per layer, dropout rate, learning rate, optimizer (Adam or Nadam), and use of bidirectionality (Aggarwal, 2024).
To avoid overfitting, dropout regularization and early stopping based on validation loss are applied during training. Once the residual model is trained, the final hybrid forecast is obtained by summing the ARIMA forecast and the LSTM-predicted residuals, thereby reconstructing the full signal. This additive combination integrates the strengths of both models and improves predictive robustness.
Model performance is assessed using standard forecast accuracy metrics: mean absolute error (MAE), root mean squared error (RMSE), and the Pearson correlation between forecasts and empirical values are checked. In addition, a rolling-origin evaluation procedure is used to simulate real-time forecasting conditions and evaluate the stability of the model across multiple forecasting windows (Hyndman and Athanasopoulos, 2018).
The hybrid model is benchmarked against standalone ARIMA, univariate LSTM, and multivariate LSTM models. This comparative analysis helps evaluate whether the hybrid strategy offers a statistically and practically significant improvement in forecasting short-term fluctuations in business sentiment indicators within the construction sector.

2.4 Multivariate long short-term memory network

In addition to univariate forecasting, this study adopts a multivariate LSTM network to model and predict business sentiment indicators in the construction sector. The multivariate approach incorporates a set of exogenous macroeconomic and sectoral indicators that are hypothesized to influence business confidence. By integrating these variables, the model captures both temporal dynamics and cross-variable dependencies that traditional time series models might overlook (Hochreiter and Schmidhuber, 1997; Zhang et al., 2003).
The multivariate LSTM model receives, at each time step, a vector of input features rather than a single scalar observation. These features include auxiliary variables from related sectors. Among them are business tendency indicators for industry, trade, and households, macroeconomic aggregates such as Polish real GDP growth. The first three indicators are derived from business tendency surveys carried out by the Research Institute for Economic Development, affiliated with the Warsaw School

of Economics. The assumption is that these indicators, though not directly part of the construction sector, contain valuable signals that improve the quality of forecasts. Before model training, all features are normalized using the z-score transformation to ensure that they share a common scale, reducing bias during optimization and accelerating convergence (Bengio, 2012). The time series is then converted into supervised learning format using a sliding window. Each training sample consists of a multivariate input matrix (timesteps $\times$ features), where the target is the value of the sentiment indicator at the next time step.

The LSTM network is trained using a backpropagation through time algorithm with mean squared error as the loss function. To optimize generalization, a set of hyperparameters is tuned, including the sequence length (number of lags), the number of neurons in the hidden layers, the number of LSTM layers, dropout regularization rates, learning rate, and the type of optimizer (Adam or Nadam) (Hochreiter and Schmidhuber, 1997; Aggarwal, 2024). Grid search is employed to explore combinations of these hyperparameters, and the best configuration is selected based on the minimum validation loss.

To avoid overfitting and improve model generalization, the training pipeline includes early stopping, whereby training halts if the validation performance does not improve over a specified number of epochs. The training and validation sets are split chronologically to reflect real-world forecasting conditions (Zhang et al., 2003).

Forecasts are generated using an autoregressive strategy for multi-step ahead prediction. In this procedure, the model's predictions are recursively fed back as inputs to forecast future periods. This technique simulates real deployment conditions where only past observations and model forecasts are available for prediction (Yan, 2024).

The multivariate LSTM model complements the univariate and hybrid architectures by expanding the available feature space and introducing exogenous explanatory variables. While more complex and computationally demanding, this architecture provides an opportunity to capture richer dynamics in the data, making it particularly useful for modeling the intricate and delayed effects of macroeconomic developments on business sentiment in the construction sector.

3 Application

3.1 Data and research design

This study utilizes data from the business tendency surveys conducted by the Research Institute of Economic Development at the Warsaw School of Economics. The dataset comprises 90 variables spanning six sectors of the Polish economy: the industry (IRGIND), construction, trade (IRGTRD), agriculture, banking (available from Q2 2022 within the financial institutions sector), and households (IRGKGD).

The business climate surveys collect qualitative data, reflecting the subjective assessments of entrepreneurs and consumers regarding ongoing economic processes. These surveys capture perceptions related to supply and demand dynamics, current business conditions, and short-term expectations. Respondents provide insights into both their specific market environments and the broader economy, enabling the monitoring of economic activity on a quarterly basis and the early identification of emerging risks. For the purpose of this study, we focus on the construction sector, leveraging its long historical time series and relevance to broader economic trends. The analysis centres on several key indicators within this sector: the economic sentiment indicator, production volume, investments, the financial standing of firms, and subjective evaluations of both the general economic outlook and the construction industry's condition. The time series used for model span from the first quarter of 1998 to the end of 2024. Additionally, a set of exogenous variables – including indicators from three sectors of the economy and GDP growth – were used as inputs in the multivariate LSTM models.

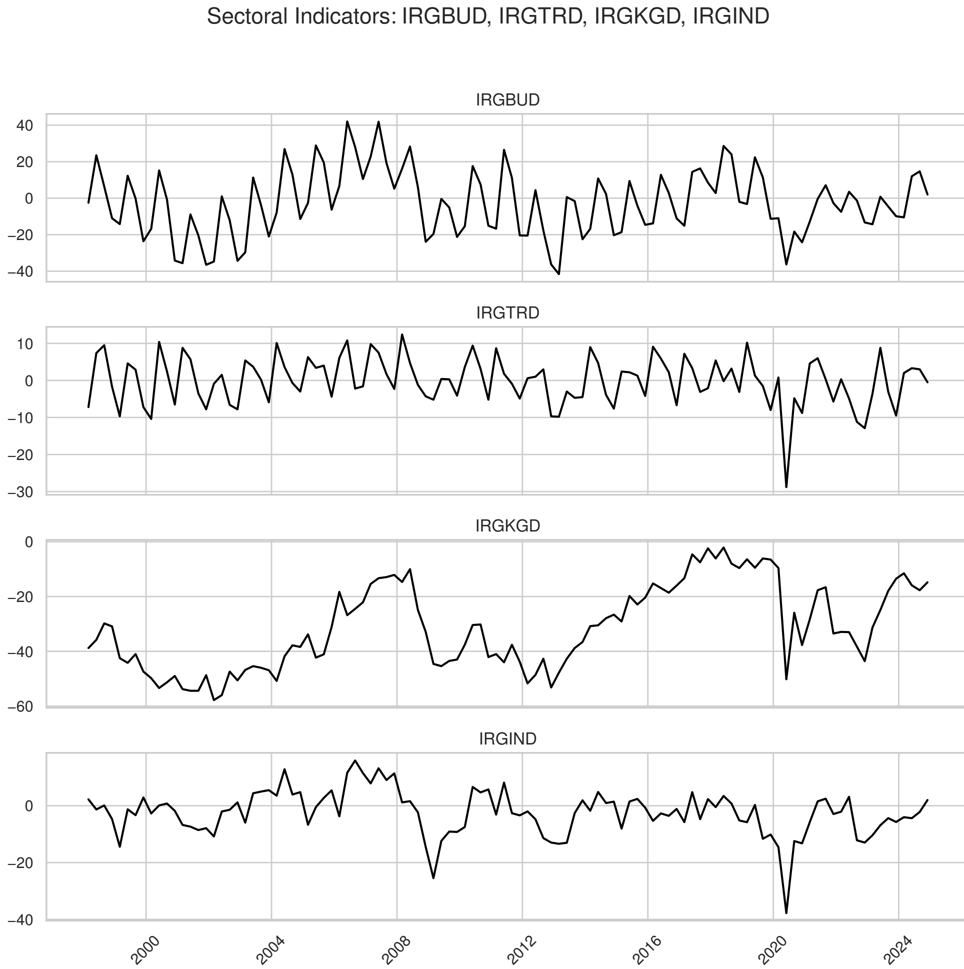
For each target variable, a univariate LSTM model was trained. Before model training, the data was normalized using standard scaling. A comprehensive grid search was performed to identify the optimal combination of hyperparameters. The search space included sequence lengths of 2, 4, 6, 8, 12 and 48 time steps; the number of LSTM neurons per layer was set to 50, 75, or 100; and model architectures with 1 stacked LSTM layer were evaluated. A fixed dropout rate of 0.1 was applied to mitigate overfitting. Additionally, learning rates of 0.01, 0.0005, and 0.0003 were tested alongside batch sizes of 16 and 32. Both the Adam and Nadam optimizers were explored. To assess the potential benefit of bidirectional sequence modeling, both unidirectional and bidirectional LSTM configurations were included in the search. Model selection was based on the lowest validation loss. In parallel, classical ARIMA models were calibrated for each variable. The order of integration was determined using the augmented Dickey-Fuller (ADF) test. The presence of seasonality was assessed via decomposition, and seasonal ARIMA specifications were applied where appropriate.

To integrate the strengths of both approaches, hybrid ARIMA-LSTM models were constructed. In these models, residuals from the ARIMA forecasts were used as input for LSTM networks, aiming to capture any remaining nonlinear patterns in the data. The same hyperparameter grid was applied to optimize the residual-based LSTM architecture.

Additionally, multivariate LSTM models were trained using the selected exogenous variables to assess the impact of broader economic conditions on each target series. After reviewing multiple feature selection strategies and evaluating alternative sets of predictors, we finalized a group of four explanatory variables to include in the multivariate LSTM model. The final selection was guided by both domain knowledge and exploratory correlation analysis i.e. IRGTRD, IRGKGD, IRGIND (see figure 2) and macroeconomic indicator GDP.

Finally, forecasts from each modeling approach – ARIMA, LSTM, hybrid ARIMA-

Figure 2: Time series of sectoral indicators from the IRG dataset: IRGBUD (Construction), IRGTRD (Trade), IRGKGD (Households), and IRGIND (Industry)



Notes: The last three variables are used as exogenous predictors in the multivariate LSTM model.

-LSTM, and multivariate LSTM – were evaluated and compared using standard metrics including MAE, RMSE and Pearson correlation.

To evaluate the robustness and practical forecasting performance of the developed models, we implement a rolling-origin forecast evaluation framework. This approach enables assessment of model stability and generalization across multiple temporal segments. The time series is partitioned into training and three sequential testing windows. For each rolling window, the model is trained on a subset of the available historical data and used to generate out-of-sample forecasts over a fixed horizon. Performance metrics are computed by comparing forecasts to the actual values in the holdout set.

This rolling-origin strategy provides a robust measure of model performance over time and helps identify whether models are sensitive to structural shifts or time-specific anomalies. By evaluating each model across multiple forecast windows, we ensure that the reported results are not biased by a single split or static holdout set, offering a more comprehensive assessment of forecast accuracy in dynamic real-world conditions. At the end final results, including model parameters, forecast accuracy, and plots, were saved for each target variable and modeling approach. All computations were performed using the Python programming language.

3.2 Stationarity and ARIMA model specification

Before fitting ARIMA-based models, each time series was tested for stationarity using ADF test. The ADF test assesses the null hypothesis that a unit root is present in the time series, implying non-stationarity. If the p-value is below a significance threshold (typically 0.05), the null hypothesis is rejected, indicating that the series is stationary. The results of the ADF tests are summarized in Table 1. Among the six variables considered, only financial situation and sector evaluation exhibit p-values below 0.05, with statistics of -3.01 and -2.91, respectively. These results suggest that these two series are stationary at the 5% significance level. The remaining variables, including investments, economic evaluation, production, and IRGBUD, have higher p-values (ranging from 0.06 to 0.26), indicating that the null hypothesis of a unit root cannot be rejected. As such, differencing was applied accordingly when specifying ARIMA models for these series.

The ARIMA order (p, d, q) and the corresponding seasonal order (P, D, Q, s) were selected based on a combination of AIC and BIC minimization criteria. For example, the best-fitting model for financial situation was determined to be ARIMA(1, 0, 0) with a seasonal component (1, 0, 2, 4), achieving an AIC of 839.35 and a BIC of 852.76. In contrast, economic evaluation required a higher-order ARIMA(2, 1, 1) with seasonal order (1, 0, 1, 4), yielding higher information criteria values (AIC = 897.39, BIC = 913.43), reflecting greater model complexity or noisier data.

The table also highlights variability in the degree of differencing (d), where differencing is needed (d=1) in series such as economic evaluation, production, and IRGBUD, but not in financial situation or sector evaluation. Seasonality was modeled uniformly with

a seasonal frequency of 4 (quarterly data), though the specific P, D, Q parameters vary across series.
These findings underscore the heterogeneity in the underlying dynamics of the economic indicators, justifying the tailored ARIMA configurations used in the forecasting experiments.

Table 1: ADF Test Results and ARIMA Specifications

Variable	ADF Stat.	p-value	ARIMA (p,d,q)	Seasonal (P,D,Q,s)	AIC	BIC
financial situation	-3.01	0.034	(1,0,0)	(1,0,2,4)	839.35	852.76
investments	-2.66	0.081	(0,1,2)	(1,0,1,4)	764.68	778.04
economic evaluation	-2.22	0.199	(2,1,1)	(1,0,1,4)	897.39	913.43
sector evaluation	-2.91	0.044	(3,0,0)	(1,0,1,4)	881.56	897.66
production	-2.06	0.261	(0,1,1)	(1,0,1,4)	854.64	865.34
IRGBUD	-2.77	0.062	(1,1,2)	(2,0,1,4)	784.87	806.25

3.3 LSTM model estimation and hyperparameter tuning

The performance of the LSTM models was evaluated using an extensive hyperparameter tuning procedure for each of the six variables under analysis. The search space for tuning was defined across several architectural and training dimensions, including sequence length, number of neurons, number of layers, dropout rate, learning rate, batch size, optimizer type, and whether bidirectional architecture was used.
This section presents the results of hyperparameter tuning for different LSTM-based architectures, including univariate LSTM, multivariate LSTM, and hybrid models combining ARIMA residuals with neural network forecasting. Each configuration was evaluated across six economic variables, with validation loss serving as the primary metric of model performance. Table 2 presents final parameters for each considered models. Chart 3 presents training and validation loss of different models for IRGBUD as an example.
Multivariate LSTM models consistently outperformed their univariate and hybrid counterparts across most variables. For instance, in the case of financial situation, the multivariate model achieved the lowest validation loss of 0.0834, outperforming the univariate LSTM (0.1047) and the hybrid (0.1070). Similarly, for investments, the multivariate model recorded a loss of 0.0698, clearly better than both the univariate (0.1082) and hybrid (0.1559) alternatives. This suggests that incorporating additional explanatory variables helps the LSTM learn broader dependencies, confirming the

Table 2: LSTM-based model configurations and validation loss per variable

Variable	Model	Seq.	Neurons	Layers	Dropout	LR	Batch	Optimizer	Val. Loss
	LSTM	4	100	2	0.1	0.0005	16	Nadam	0.1047
fin. situation	Multivariate	4	100	1	0.1	0.0005	16	Nadam	0.0834
	Hybrid	2	75	1	0.1	0.0005	32	Nadam	0.1070
	LSTM	6	75	2	0.1	0.0005	16	Nadam	0.1082
investments	Multivariate	6	100	1	0.1	0.0005	16	Nadam	0.0698
	Hybrid	2	50	1	0.1	0.0005	32	Adam	0.1559
	LSTM	6	100	1	0.1	0.0005	16	Nadam	0.1005
econ. eval.	Multivariate	2	100	1	0.1	0.0005	16	Nadam	0.0786
	Hybrid	2	75	1	0.1	0.0003	32	Nadam	0.1796
	LSTM	4	100	1	0.1	0.0005	16	Nadam	0.1185
sector eval.	Multivariate	2	50	1	0.1	0.0005	16	Nadam	0.0825
	Hybrid	2	75	1	0.1	0.0005	32	Nadam	0.1465
	LSTM	12	75	1	0.1	0.0100	32	Nadam	0.0691
production	Multivariate	12	75	3	0.1	0.0100	16	Adam	0.0657
	Hybrid	2	100	1	0.1	0.0100	32	Nadam	0.1135
	LSTM	4	50	1	0.1	0.0005	16	Adam	0.2847
IRGBUD	Multivariate	4	100	1	0.1	0.0005	16	Nadam	0.1933
	Hybrid	4	100	1	0.1	0.0005	32	Adam	0.1695

benefits of multivariate settings noted in prior literature (Ahmed et al., 2010; Hyndman and Athanasopoulos, 2018).

The hybrid models, which use residuals from ARIMA as inputs to the LSTM, showed some promise, particularly in the case of IRGBUD, where the hybrid configuration performed better (validation loss = 0.1695) than the standalone univariate LSTM (0.2847). This supports findings from recent studies emphasizing that hybrid architectures can be particularly effective when linear and nonlinear components coexist (Zhang et al., 2003; Michańków et al., 2024).

Regarding model structure, most effective configurations used a single LSTM layer, aligning with findings that deeper architectures do not always improve performance for time series data (Hochreiter and Schmidhuber, 1997). The number of neurons ranged from 50 to 100. Larger neuron counts slightly improved performance in multivariate settings but added complexity and training time. The dropout rate was fixed at 0.1 for all experiments, a standard value to mitigate overfitting while retaining model capacity (Bishop, 1995).

Sequence length, which defines how many past time steps the model considers, was another important factor. Shorter sequences (length = 2 or 4) performed better for variables with less persistent dynamics (e.g., sector evaluation), while longer sequences (length = 6 or 12) were more effective for variables like production or investments, which may rely on extended memory to capture cyclical trends. This corroborates earlier observations on memory depth and temporal dependencies in LSTM design (Hochreiter and Schmidhuber, 1997; Tsay, 2005).

Optimizer choice also played a role in performance. Most top-performing models used the Nadam optimizer, a blend of RMSProp and Nesterov momentum, known for its ability to accelerate convergence in recurrent networks (Dixon et al., 2020). In contrast, Adam – a popular general-purpose optimizer – was used in a few cases, particularly in hybrid models. Adam's effectiveness in hybrid settings may be due to its adaptiveness to the less predictable residual structure (Hia et al., 2023).

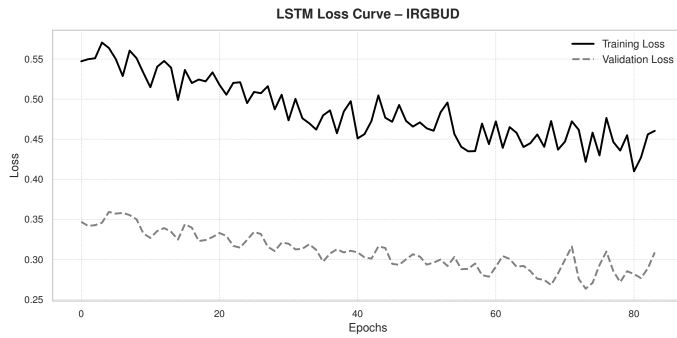
Batch size was alternated between 16 and 32. Smaller batches (size = 16) slightly improved performance by offering more frequent weight updates, but larger batches were beneficial in hybrid setups, likely due to more stable gradient estimates (Bengio, 2012).

Overall, these findings underscore the value of thorough hyperparameter optimization and the complementary nature of hybrid models. While multivariate LSTM remains the most consistent across variables, hybrid models can improve robustness when the underlying data exhibit both linear and nonlinear characteristics.

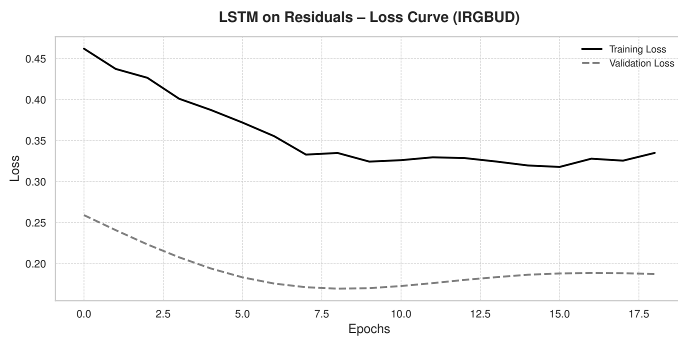
3.4 Model performance evaluation

This section presents a comparative evaluation of four forecasting models – ARIMA, LSTM, ARIMA+LSTM hybrid, and multivariate LSTM – across six business sentiment indicators: economic evaluation, financial situation, investments, IRGBUD, production, and sector evaluation. The performance metrics considered include MAE,

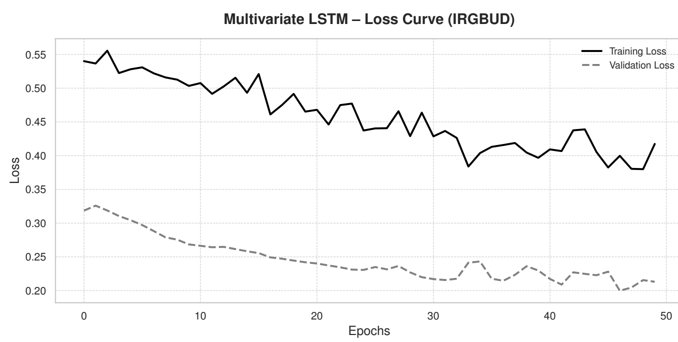
Figure 3: Loss curves for IRGBUD



(a) LSTM



(b) ARIMA+LSTM Hybrid



(c) Multivariate LSTM

RMSE, and Pearson correlation coefficient, computed for both the testing period and predictions using three rolling forecast windows (W1, W2, and W3). Table 3 presents the statistics. Charts 4-6 presents calculated forecasts.

In the evaluation on the testing dataset, traditional ARIMA and hybrid ARIMA+LSTM models consistently achieved the lowest MAE and RMSE values across most variables. For instance, ARIMA and the hybrid model performed best on economic evaluation, investments, and IRGBUD, with MAE values below 8 and high correlation scores (0.56-0.73). The hybrid model generally offered slight improvements over ARIMA in capturing variance, suggesting its ability to model residual nonlinearities not captured by the linear ARIMA component.

LSTM-only models showed higher testing error in some cases (e.g., economic evaluation, MAE = 18.26) but also demonstrated strong correlation in areas like production (0.81) and sector evaluation (0.44), hinting at their potential for nonlinear pattern learning. However, their variance remained higher compared to ARIMA-based models.

The multivariate LSTM models, although theoretically advantageous due to the inclusion of exogenous variables, underperformed in the testing period in most categories, with significantly higher MAE and RMSE (e.g., production, MAE = 26.26; RMSE = 32.03). This suggests overfitting or challenges in capturing the complex interactions between variables with limited data.

Forecasting accuracy was tested across three rolling windows (W1-W3), simulating real-world forecasting conditions.

Rolling-W1: In the short-term forecasts, ARIMA and ARIMA+LSTM hybrid models delivered the most reliable and accurate predictions. For example, in the financial situation variable, the hybrid model achieved an MAE of 2.62 with a high correlation of 0.97. Similar results were noted for IRGBUD and production. Notably, the hybrid models slightly outperformed ARIMA in most cases, validating the value of modeling residuals using LSTM. In contrast, standalone LSTM models performed inconsistently in W1, with deteriorated correlation in economic evaluation (-0.63) and investments (0.10), while showing competitive accuracy for production (correlation = 0.99).

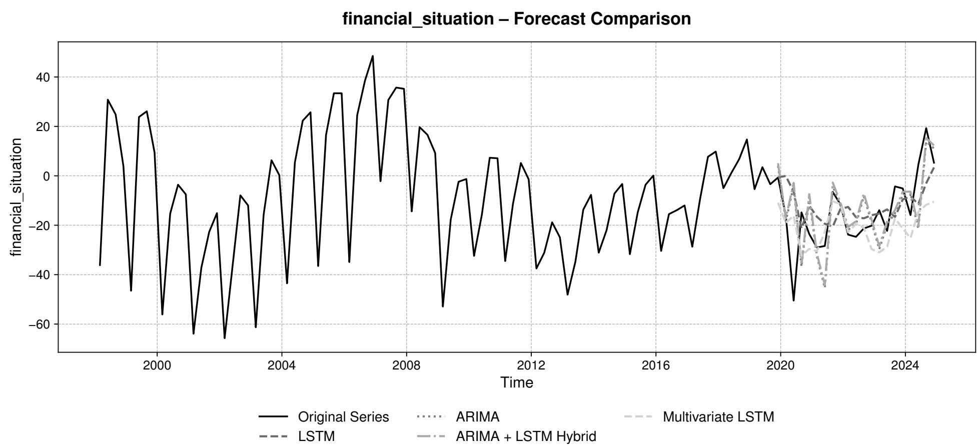
Rolling-W2: As the forecast horizon extended, hybrid models maintained robustness, outperforming LSTM and multivariate LSTM in most indicators. Multivariate LSTM again showed weaker performance, with low or negative correlations for several indicators, including economic evaluation (-0.37) and financial situation (-0.59), despite modest error values.

Rolling-W3: For longer-term forecasts, hybrid models continued to outperform both standalone LSTM and multivariate LSTM models in terms of MAE and RMSE. LSTM models occasionally yielded high correlation (e.g., economic evaluation, 0.80), but also showed volatility in performance, such as in investments (-0.70). Multivariate LSTM generally underperformed across variables, indicating overfitting or an inability to

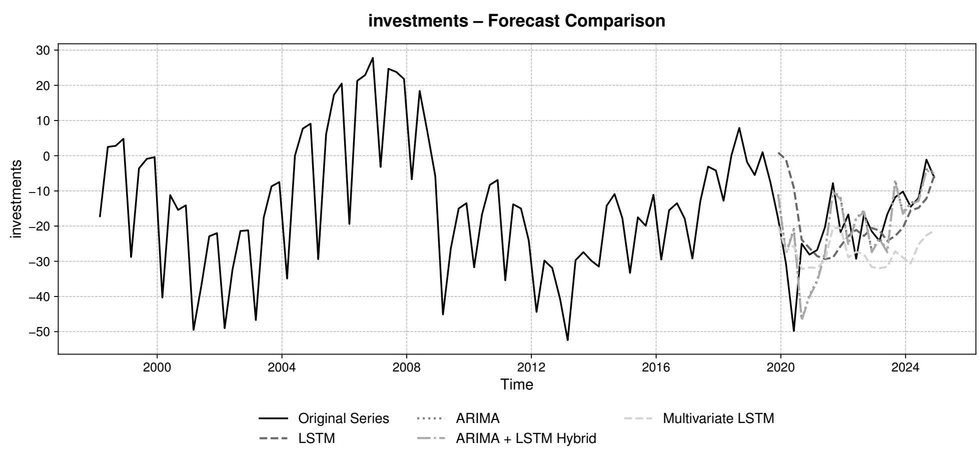
Table 3: Forecast accuracy comparison across models and forecasting windows

Variable	Model	In-sample			Rolling-W1			Rolling-W2			Rolling-W3		
		MAE	RMSE	Corr.	MAE	RMSE	Corr.	MAE	RMSE	Corr.	MAE	RMSE	Corr.
economic evaluation	ARIMA	14.43	20.61	0.59	10.56	12.44	0.79	16.90	18.71	0.57	32.44	42.94	0.21
	ARIMA+LSTM Hybrid	14.82	20.69	0.59	9.83	11.58	0.79	17.55	19.61	0.59	32.49	42.96	0.21
	LSTM	18.26	23.13	0.45	18.48	21.71	-0.63	23.37	25.90	-0.82	47.28	55.42	0.80
	Multivariate LSTM	29.75	34.55	0.64	16.29	21.32	-0.29	21.18	22.93	-0.37	21.02	23.13	0.04
financial situation	ARIMA	10.13	14.77	0.48	2.49	2.99	0.97	11.45	12.75	0.60	22.94	27.79	0.16
	ARIMA+LSTM Hybrid	10.29	14.88	0.48	2.62	3.11	0.97	11.70	12.99	0.60	22.70	27.50	0.19
	LSTM	10.51	15.01	0.20	9.55	11.95	0.42	15.00	17.86	0.01	28.57	32.98	-0.50
	Multivariate LSTM	9.95	13.20	0.48	26.08	30.67	0.56	20.15	24.38	-0.59	28.64	30.17	-0.10
invest- -ments	ARIMA	7.36	10.10	0.56	12.48	13.11	0.90	9.00	11.04	0.63	16.70	20.23	0.20
	ARIMA+LSTM Hybrid	7.37	10.11	0.56	12.67	13.29	0.90	9.13	11.16	0.63	16.58	20.09	0.20
	LSTM	7.91	12.29	0.09	10.47	12.27	0.10	5.64	7.10	0.75	22.78	26.82	-0.70
	Multivariate LSTM	10.93	12.80	0.21	13.53	17.48	-0.58	13.71	15.86	-0.20	19.90	21.30	-0.41
IRGBUD	ARIMA	7.23	13.27	0.40	5.49	7.38	0.95	11.38	14.12	0.76	15.42	22.41	0.21
	ARIMA+LSTM Hybrid	7.23	13.34	0.40	5.44	7.49	0.95	11.45	14.14	0.76	15.74	22.73	0.19
	LSTM	8.38	11.03	0.35	11.07	12.61	0.24	16.91	20.42	-0.35	12.22	14.92	0.71
	Multivariate LSTM	6.99	10.16	0.55	14.81	17.48	-0.66	13.38	15.27	0.33	16.08	19.05	-0.30
production	ARIMA	8.47	11.39	0.89	8.38	9.96	0.99	13.81	16.56	0.93	22.72	31.06	0.68
	ARIMA+LSTM Hybrid	8.39	11.33	0.89	8.46	10.01	0.99	13.78	16.64	0.93	22.93	31.13	0.68
	LSTM	8.98	10.92	0.81	19.66	20.15	0.99	14.37	16.60	0.90	26.62	35.30	0.67
	Multivariate LSTM	26.26	32.03	0.41	34.03	42.18	0.24	38.49	42.44	-0.32	38.64	41.45	0.10
sector evaluation	ARIMA	10.92	15.54	0.73	7.47	9.35	0.86	26.58	30.35	0.19	24.12	31.60	0.21
	ARIMA+LSTM Hybrid	10.85	15.85	0.73	7.40	9.29	0.85	26.91	30.75	0.20	24.16	31.58	0.21
	LSTM	16.18	19.44	0.44	33.11	35.82	-0.41	23.61	25.71	-0.59	30.88	35.18	0.64
	Multivariate LSTM	18.48	22.84	0.70	11.29	16.60	0.25	13.50	15.61	0.10	24.51	26.63	0.45

Figure 4: Forecasting performance visualizations for six economic indicators

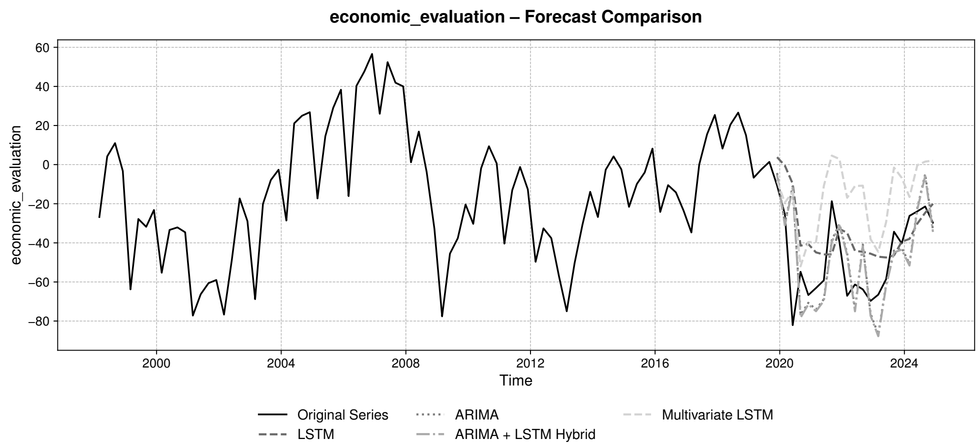


(a) Financial Situation

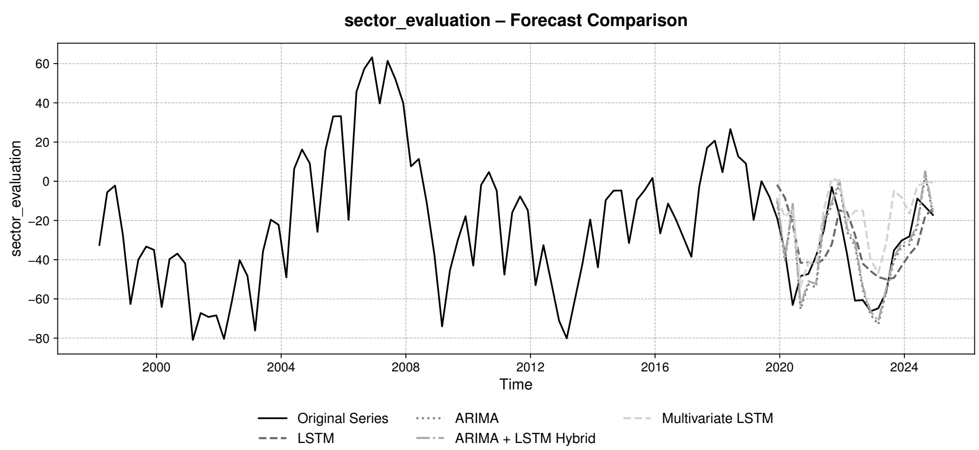


(b) Investments

Figure 5: Forecasting performance visualizations for six economic indicators

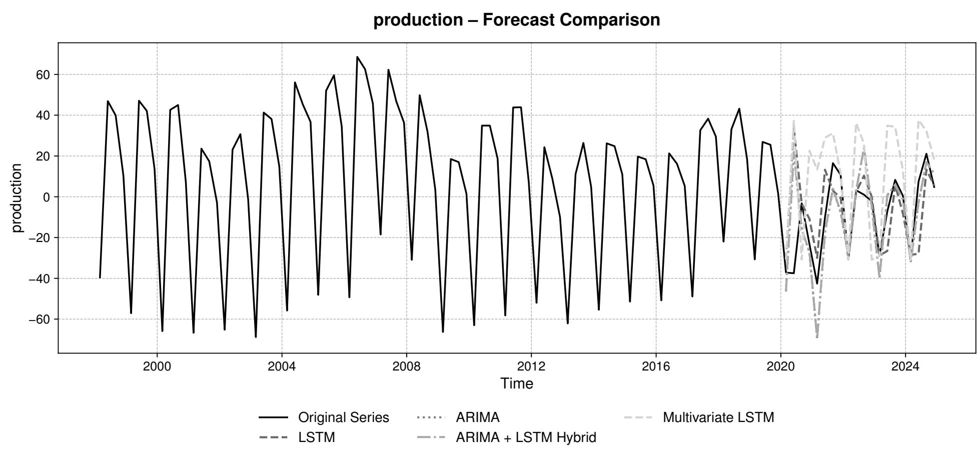


(a) Economic Evaluation

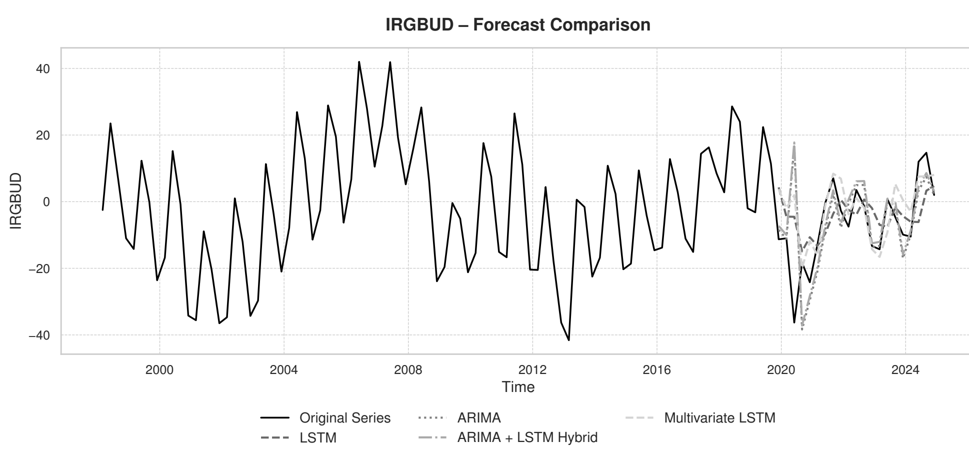


(b) Sector Evaluation

Figure 6: Forecasting performance visualizations for six economic indicators



(a) Production



(b) IRGBUD

generalize well to future data, possibly due to the noisy influence of external inputs or insufficient data for training.
ARIMA+LSTM Hybrid consistently provided balanced performance across all evaluation metrics and forecast windows, confirming its value in capturing both linear and nonlinear structures of the data. LSTM models showed competitive performance in select areas, especially where nonlinearity and long-term dependencies were pronounced (e.g., production, economic evaluation), but their instability across forecast horizons limits their standalone reliability. Multivariate LSTM, despite its theoretical advantages, underperformed in this study, suggesting the need for more sophisticated input preprocessing, dimensionality reduction (e.g., PCA), or larger datasets. Overall, ARIMA and hybrid ARIMA+LSTM models offer a practical and effective approach for business sentiment forecasting in the construction sector, balancing interpretability, accuracy, and robustness across multiple forecast windows.

4 Sensitivity analysis of LSTM hyperparameters

To validate the robustness of the LSTM model configuration selected through the primary grid search, a targeted sensitivity analysis was conducted. The goal was to test whether the best configuration lies within a stable region of the hyperparameter space or whether small perturbations lead to significantly degraded performance. Table 4 and charts 7-8 present the results.
The initial grid search selected a unidirectional LSTM model with a sequence length of 4, 50 or 100 neurons, 1 hidden layer, a dropout rate of 0.1, a learning rate of 0.0005, a batch size of 16, and either the Adam or Nadam optimizer. For the sensitivity analysis, we designed a focused grid that made local modifications to the best-performing setup to examine how each component contributes to model accuracy.
The following hyperparameter dimensions were tested:
The sequence length was adjusted to 3 and 5 to investigate the model's sensitivity to slightly shorter or longer temporal memory. While the full grid had broader values, this refinement narrowed the focus around the optimum.
A two-layer architecture was evaluated instead of the originally selected single-layer network, to determine whether deeper structures might marginally enhance performance, especially in more complex multivariate configurations.
The number of neurons was fixed at 75, a value that consistently balanced performance and training efficiency across variables.
To test regularization effects, dropout rates of 0.05 and 0.15 were applied. These values were chosen to assess whether slightly lower or higher dropout would lead to underfitting or overfitting, respectively, in the context of economic time series.
Learning rate sensitivity was tested using 0.0003 and 0.0008. These values bracket the optimal 0.0005, allowing insight into training dynamics under slightly slower or faster convergence conditions.
Batch sizes of 8 and 32 were used to test sensitivity to mini-batch granularity. Smaller

batch sizes can provide more stable generalization at the cost of longer training times, while larger batches offer efficiency gains but may risk convergence to sharper minima. Both Adam and Nadam optimizers were reevaluated, as they had previously shown good performance, to confirm their consistency under modified configurations.
Finally, bidirectional LSTM architectures were tested to determine whether introducing forward and backward sequence processing could enhance forecasting accuracy, even though they were not selected as optimal in the initial grid.
The following targeted variations were applied, producing seven configurations (sensitivity 1 to sensitivity 7):

sensitivity 1: Sequence length varied to 3 and 5 (vs. optimal 4)

sensitivity 2: Reduced number of neurons to 75

sensitivity 3: Increased number of layers to 2

sensitivity 4: Dropout adjusted to 0.05 and 0.15

sensitivity 5: Learning rate tuned to 0.0003 and 0.0008

sensitivity 6: Batch sizes tested at 8 and 32

sensitivity 7: Enabled bidirectional LSTM (BiLSTM)

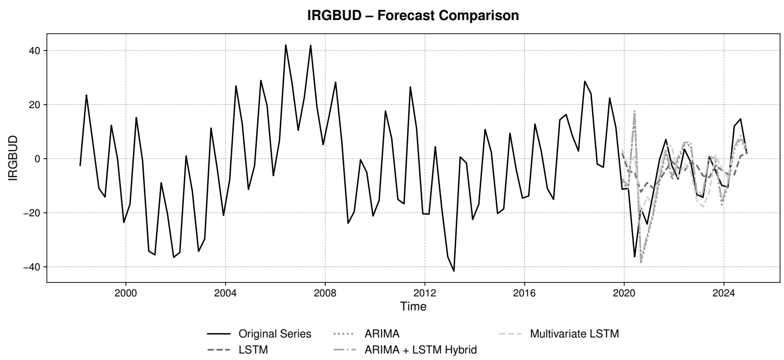
The results demonstrate that the ARIMA+LSTM Hybrid model remains the most stable and accurate across parameter variations, confirming its robustness. It consistently performs well under different configurations, outperforming both the standalone LSTM and the multivariate LSTM.
Sequence length adjustments (sensitivity 1) showed that even small deviations from the optimal value can influence the model's ability to capture temporal dependencies, particularly affecting standard LSTM and Multivariate models.
Reducing the number of neurons (sensitivity 2) had limited effect on the hybrid model, suggesting that it is less sensitive to this parameter. However, other architectures displayed small increases in error, indicating a trade-off between model complexity and generalization.
Increasing the number of layers (sensitivity 3) resulted in poorer overall performance, likely due to overfitting or training instability. This finding suggests that deeper networks may not yield better results in the context of macroeconomic data.
Dropout modifications (sensitivity 4) caused minor fluctuations in error, but no consistent improvement. The models appeared tolerant to slight changes in regularization, reinforcing that the original dropout value was well-calibrated.
Learning rate variation (sensitivity 5) confirmed that the chosen value (0.0005) was already optimal. Other tested values either slowed convergence or led to unstable learning without significant gains in accuracy.
Batch size (sensitivity 6) changes showed that the models were largely insensitive to this parameter, with only marginal differences in performance across the range tested.

Enabling bidirectionality (sensitivity 7) did not lead to meaningful gains. While it theoretically allows the network to learn from future and past time steps, it did not improve accuracy in this application, suggesting that unidirectional LSTM is more appropriate for economic forecasting where time flows forward naturally.
Overall, the analysis confirms that the ARIMA+LSTM hybrid model is robust to reasonable hyperparameter variation. While certain parameters like sequence length and depth have noticeable effects, others such as dropout, batch size, and learning rate can be adjusted more flexibly without substantial impact on performance.

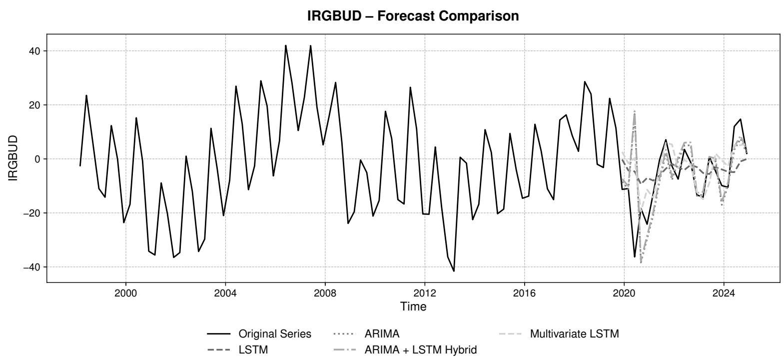
Table 4: IRGBUD forecast accuracy comparison for sensitivity configurations.

Type	Model	MAE	RMSE	Corr.
estimation	LSTM	8.38	11.03	0.35
	ARIMA+LSTM Hybrid	7.23	13.34	0.40
	Multivariate LSTM	6.99	10.16	0.55
sensitivity 1	LSTM	8.67	11.11	0.37
	ARIMA+LSTM Hybrid	7.22	13.34	0.40
	Multivariate LSTM	7.08	10.28	0.52
sensitivity 2	LSTM	9.80	12.51	0.42
	ARIMA+LSTM Hybrid	7.21	13.29	0.40
	Multivariate LSTM	6.87	9.93	0.56
sensitivity 3	LSTM	11.24	14.31	0.28
	ARIMA+LSTM Hybrid	7.22	13.32	0.40
	Multivariate LSTM	10.39	13.35	0.52
sensitivity 4	LSTM	8.25	10.89	0.39
	ARIMA+LSTM Hybrid	7.22	13.32	0.40
	Multivariate LSTM	6.93	9.95	0.56
sensitivity 5	LSTM	8.55	10.93	0.41
	ARIMA+LSTM Hybrid	7.22	13.31	0.40
	Multivariate LSTM	6.88	9.77	0.58
sensitivity 6	LSTM	8.78	11.13	0.39
	ARIMA+LSTM Hybrid	7.22	13.33	0.40
	Multivariate LSTM	7.05	10.14	0.56
sensitivity 7	LSTM	8.17	10.79	0.40
	ARIMA+LSTM Hybrid	7.22	13.29	0.40
	Multivariate LSTM	7.05	10.11	0.55

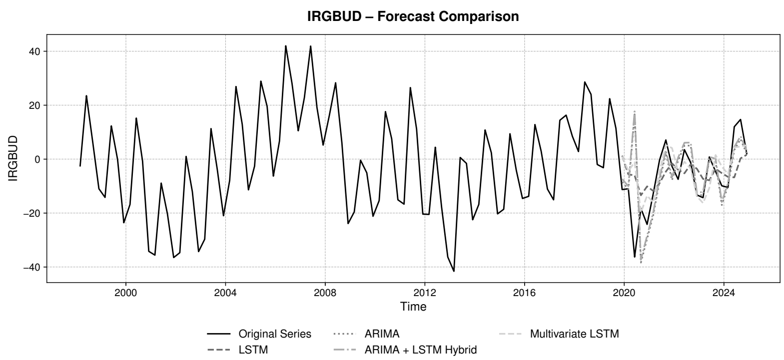
Figure 7: Comparison of IRGBUD model results for sensitivity analysis.



(a) Sensitivity 1

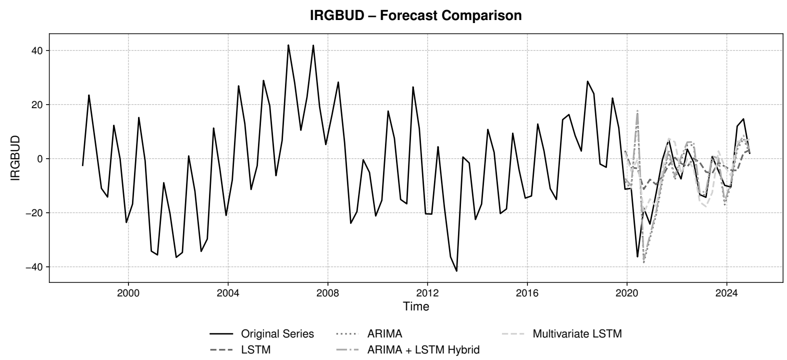


(b) Sensitivity 2

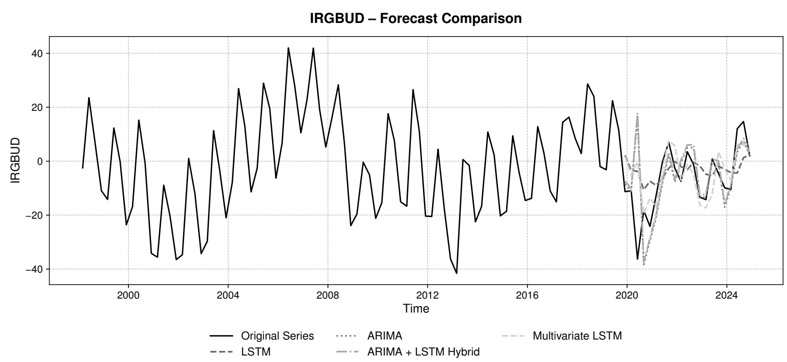


(c) Sensitivity 4

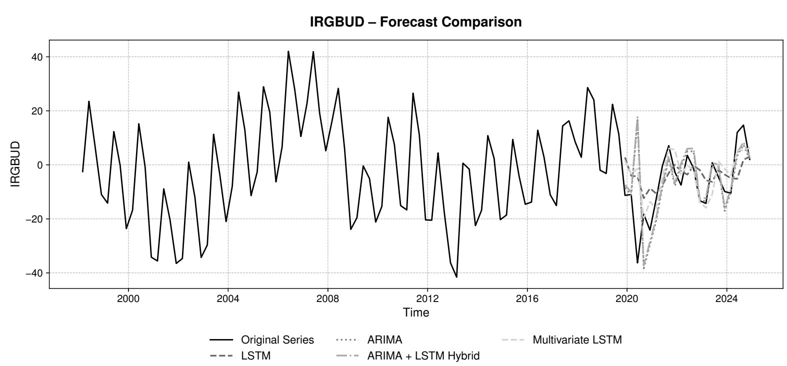
Figure 8: Comparison of IRGBUD model results for sensitivity analysis.



(a) Sensitivity 5



(b) Sensitivity 6



(c) Sensitivity 7

5 Model enhancement and future directions

Although traditional ARIMA and hybrid ARIMA+LSTM models demonstrated the highest forecasting accuracy in this study, there remains substantial potential for enhancing deep learning-based approaches. Future research may consider the following directions to further improve model performance:

1. Feature engineering and selection: incorporating lagged variables, differencing, moving averages, or applying dimensionality reduction techniques such as principal component analysis (PCA) (Jolliffe, 2002) could help reduce noise and improve the relevance of input features in multivariate models These enhancements represent promising directions for improving time series forecasting models and can contribute to more accurate and interpretable results in future studies.

2. Advanced LSTM architectures: more sophisticated recurrent neural network structures, such as attention-based LSTM models (Vaswani et al., 2017), may enhance the capacity to capture complex temporal dependencies present in macroeconomic time series.

3. Variable importance analysis: methods such as SHAP (shapley additive explanations) (Lundberg and Lee, 2017) or permutation importance (Breiman, 2001) can be utilized to evaluate the impact of individual exogenous variables – such as GDP, consumer confidence, or sectoral indicators – on model predictions. This analysis can guide more targeted feature selection in future model iterations.

4. Expanded sensitivity and robustness testing: beyond conventional hyperparameter tuning, future studies should implement more granular sensitivity analyses across forecast horizons and noise levels.

5. Alternative machine learning benchmarks: additional models like support vector regression (SVR) and random forests could offer valuable benchmarks, particularly for non-sequential prediction tasks and recommend them for future comparative studies.

In conclusion, while ARIMA-based models currently offer strong baseline performance for forecasting economic indicators derived from business survey data, incorporating modern deep learning architectures and multivariate inputs presents a promising avenue for future research. A systematic approach to feature engineering, architecture optimization, and interpretability will be essential for advancing the accuracy and trustworthiness of machine learning in economic forecasting.

References

[1] Aggarwal S., (2024), The ultimate guide to building your own lstm models, accessed: 2024-03-21.

[2] Ahmed N. K., Atiya A. F., Gayar N. E., El-Shishiny H., (2010), An empirical comparison of machine learning models for time series forecasting, *Econometric Reviews* 29(5-6), 594–621.

[3] Baranochnikov I., Ślepaczuk R., (2023), A comparison of long short-term memory and gated recurrent unit models' architectures with novel walk-forward approach to algorithmic investment strategy, November 9, 2023.

[4] Bengio Y., (2012), Practical recommendations for gradient-based training of deep architectures, [in:] *Neural Networks: Tricks of the Trade*, [eds.:] G. Montavon, G. B. Orr, K.-R. Müller, volume 7700 of Lecture Notes in Computer Science, Springer, 437–478.

[5] Bishop C. M., (1995), *Neural Networks for Pattern Recognition*, Oxford University Press.

[6] Box G. E. P., Jenkins G. M., Reinsel G. C., Ljung G. M., (2015), *Time Series Analysis: Forecasting and Control*, Wiley.

[7] Breiman L., (2001), Random forests, *Machine Learning* 45(1), 5–32.

[8] Dixon M. F., Halperin I., Bilokon P., (2020), *Machine Learning in Finance: From Theory to Practice*, Springer.

[9] GeeksforGeeks, (2024), What is lstm – long short term memory?, accessed: 2024-06-10.

[10] Hamid S. H., Iqbal Z., (2004), Using neural networks for forecasting volatility of s&p 500 index futures prices, *Journal of Business Research* 57(10), 1116–1125.

[11] Hia S., Kuswanto H., Prastyo D. D., (2023), Robustness of support vector regression and random forest models: A simulation study, [in:] *Data Science and Emerging Technologies. DaSET 2022*, [eds.:] Y. B. Wah, M. W. Berry, A. Mohamed, D. Al-Jumeily, volume 165 of Lecture Notes on Data Engineering and Communications Technologies, Springer, Singapore.

[12] Hochreiter S., Schmidhuber J., (1997), Long short-term memory, *Neural Computation* 9(8), 1735–1780.

[13] Hyndman R. J., Athanasopoulos G., (2018), *Forecasting: Principles and Practice*, OTexts, Melbourne, Australia, 3rd edition.

[14] Jolliffe I. T., (2002), *Principal Component Analysis*, Springer.

[15] Kashif K., Ślepaczuk R., (2024), Lstm-arima as a hybrid approach in algorithmic investment strategies, available at `https://arxiv.org/abs/2406.18206v1`.

[16] Lundberg S. M., Lee S.-I., (2017), A unified approach to interpreting model predictions, [in:] *Advances in Neural Information Processing Systems (NeurIPS)*.

[17] Michańków J., Sakowski P., Ślepaczuk R., (2022), The comparison of lstm in algorithmic investment strategies on btc and sp500 index, *Sensors* 22, 917.

[18] Michańków J., Sakowski P., Ślepaczuk R., (2024), Hedging properties of algorithmic investment strategies using lstm and arima-garch models for equity indices, [in:] *Proceedings of the 32nd International Conference on Information Systems Development (ISD 2024)*.

[19] Moshiri S., Cameron N., Scuse D., (1999), Static, dynamic, and hybrid neural networks in forecasting inflation, *Computational Economics* 14(3), 219–235.

[20] Roszyk N., Ślepaczuk R., (2022), The hybrid forecast of s&p 500 volatility ensembled from vix, garch and lstm models, Working Papers, Faculty of Economic Sciences, University of Warsaw, 2024–13.

[21] Taylor J. W., Buizza R., (2003), Using weather ensemble predictions in electricity demand forecasting, *International Journal of Forecasting* 19(1), 57–70.

[22] Tsay R. S., (2005), *Analysis of Financial Time Series*, Wiley&Sons, Chicago.

[23] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser L., Polosukhin I., (2017), Attention is all you need, [in:] *Advances in Neural Information Processing Systems (NeurIPS)*.

[24] Yan S., (2024), Understanding lstm and its diagrams, accessed: 2024-07-15.

[25] Zhang G., Patuwo B. E., Hu M. Y., (2003), Time series forecasting using a hybrid arima and neural network model, *Neurocomputing* 50, 159–175.

[26] Zhang G. P., (2004), *Neural Networks in Business Forecasting*, Information Science Publishing.

[27] Zhong J., Enke D., (2017), Predicting the daily return direction of the stock market using hybrid machine learning algorithms, *Financial Innovation* 3(1), 1–23.