Journal of Theoretical and Applied Information Technology
31st July 2025. Vol.103. No.14

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

UTILIZING AZURE AUTOMATED MACHINE LEARNING FOR SALES PREDICTION

HADI SYAHRIAL[1], FIKRI SALAM[2], TANTY OKTAVIA[3]

[1,2, 3] Information System Management Department, BINUS
Graduate Program Master of Information System Management,
Bina Nusantara University, Jakarta, Indonesia

E-mail: [1]hadi.syahrial@binus.ac.id; [2]fikri.salam@binus.ac.id; [3]toktavia@binus.edu

ABSTRACT

This research assesses the use of Azure Automated Machine Learning (Azure AutoML) at Company XYZ, a prominent flour manufacturer in Indonesia, to overcome the constraints of traditional forecasting techniques. Utilizing the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology and Azure AutoML's no-code architecture, predictive models were constructed employing five years of historical data. The Voting Ensemble model proved to be optimum, with the Normalized Mean Absolute Error (NMAE) surpassing the Normalized Root Mean Squared Error (NRMSE), attaining an NMAE of 0.21278 in training and enhancing to 0.12118 in testing. The Relative Deviation Averages (RDA) for Products K and Y during one semester were decreased to 10.18% and 0.76%, respectively, surpassing traditional approaches characterized by significant variability. To ascertain dependability, predictions were juxtaposed with actual sales data over a six-month period using semester-based RDA calculations, yielding findings that demonstrated considerable improvement over traditional techniques. Notwithstanding its benefits, Azure AutoML encountered constraints in automated preprocessing activities, necessitating user intervention using Microsoft SQL Server for data cleansing and preparation. The no-code interface allowed non-expert users to deploy models inside Company XYZ's Microsoft environment; nonetheless, successful implementation required a fundamental understanding of statistics and preprocessing methods, including outlier identification, as well as proficiency in MS SQL Server (Transact-SQL). This paper presents a scalable system for improving prediction accuracy via the integration of CRISP-DM methodology, SQL preprocessing, and Azure AutoML, hence facilitating AI deployment in resource-limited settings.

Keywords: *Automated Machine Learning, Azure AutoML, CRISP-DM, Sales Prediction, Time Series*

1. INTRODUCTION

Sales prediction is an essential procedure in company management, allowing organizations to make strategic choices about production, distribution, and resource allocation. Nevertheless, several organizations, including Company XYZ, continue to depend on traditional approaches that encounter considerable constraints in accurately collecting the intricate and evolving patterns of sales data [1]. Company XYZ, a prominent flour manufacturer in Indonesia, now employs manual spreadsheet-based techniques for predicting every six months.

The six-month forecast timeframe was selected for its equilibrium between responsiveness to market dynamics and meticulous planning, allowing organizations to swiftly adjust to demand variations and make more precise strategic choices compared to longer-term estimates, such as yearly forecasts. Nevertheless, the corporation encounters substantial difficulties owing to considerable discrepancies in semester forecasts. These challenges have compelled the organization to pursue a forecasting strategy to enhance strategic planning and optimize inventory.

Oversee management, guarantee seamless raw material supply chains, and enhance overall operations efficiency. Progress in artificial intelligence and machine learning provide considerable opportunities to tackle these issues. Machine learning enables computers to derive insights from previous data and discern intricate patterns with little operator oversight. Nonetheless, conventional machine learning application requires significant technical proficiency, including programming capabilities and hyperparameter optimization, which presents obstacles for organizations with constrained data science personnel. This study investigates the use of

Automated Machine Learning (AutoML), with a particular focus on Azure AutoML, to address this need. AutoML automates several technical procedures in machine learning model building, including algorithm selection and model validation, enabling non-expert users to successfully use this technology [3]. Azure AutoML provides a no-code graphical interface that enables individuals lacking extensive technical knowledge to construct machine learning models, while also allowing for easy integration with Company XYZ's Microsoft-centric IT environment to assure compatibility and operational efficiency.

Nonetheless, significant problems exist in the implementation of AutoML. Research has shown that a major challenge encountered by AutoML users is the automation of data preparation tasks without human involvement. Moreover, the integration of data from diverse sources, management of absent values, feature engineering, and data cleansing continue to pose considerable challenges in the automated preprocessing pipeline[5].

The main aims are three in number: (1) To assess the effectiveness of Azure AutoML in improving sales forecast precision relative to traditional methods by constructing the optimal model utilizing five years of historical data for two primary products (Product K and Product Y), employing Normalized Root Mean Squared Error (NRMSE) and Normalized Mean Absolute Error (NMAE) as key metrics [6], and subsequently validating the predictive reliability of the optimal model through a six-month (one semester) actual data benchmark by reproducing the superior performance metrics and predictions.CSV outputs for comparison study between the optimal model's predictions and actual data; (2) to explore automated data preparation as a significant problem in AutoML implementation by analyzing Azure.

Evaluate AutoML's inherent preprocessing functionalities and investigate alternative methods through MS SQL Server to tackle preprocessing issues; and (3) assess the efficacy of the no-code methodology in enabling non-expert users to deploy machine learning technology while examining the viability of incorporating Azure AutoML into Company XYZ's current Microsoft-centric IT infrastructure.

This research enhances academic discourse by (1) presenting concrete evidence of Azure AutoML's constraints in automated data preparation inside no-code frameworks; (2) broadening the assessment of industry-specific AutoML to the FMCG sector via case studies on flour products;

Validating an integration framework that combines the CRISP-DM methodology, Azure AutoML, and MS SQL Server preprocessing (if necessary) to improve forecasting accuracy; and evaluating non-expert accessibility to advanced AI technologies via no-code solutions while delineating requisite competency levels.

2. THEORITICAL BACKGROUND

Machine learning (ML) and automated machine learning (AutoML) have arisen as potent strategies to overcome the constraints of conventional statistical techniques. Rohaan et al. proved that supervised learning algorithms, including XGBoost, significantly enhance sales forecasting precision in B2B contexts by adeptly collecting intricate sales patterns[7]. Pavlyshenko similarly highlighted the advantages of tree-based algorithms and ensemble techniques in improving prediction performance for sales time series data[8]. Westergaard et al. assessed many AutoML systems, such as AutoGluon and PyCaret, for time series forecasting. Their results indicated that AutoML technologies consistently predict data with minimum human intervention, making them optimal for companies using constrained technological resources [9]. This corresponds with the present study's emphasis on Azure AutoML's no-code methodology to enable non-data scientist staff at Company XYZ.

Tarallo et al. examined machine learning applications in fast-moving consumer goods sales forecasting, indicating its superiority over conventional approaches in inventory management and waste reduction [10]. Nonetheless, targeted research on the efficacy of Azure AutoML in this area is limited.

3. METHODOLOGY

The study technique employs the CRISP-DM framework (Cross-Industry Standard Process for Data Mining), offering a systematic approach to data mining initiatives. The CRISP-DM framework has six phases: business understanding, data understanding, data preparation, modeling, assessment, and deployment. This technique guarantees systematic advancement throughout the project lifespan. The framework organizes the process into six discrete stages, as seen in Figure 1

Journal of Theoretical and Applied Information Technology
31st July 2025. Vol.103. No.14

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

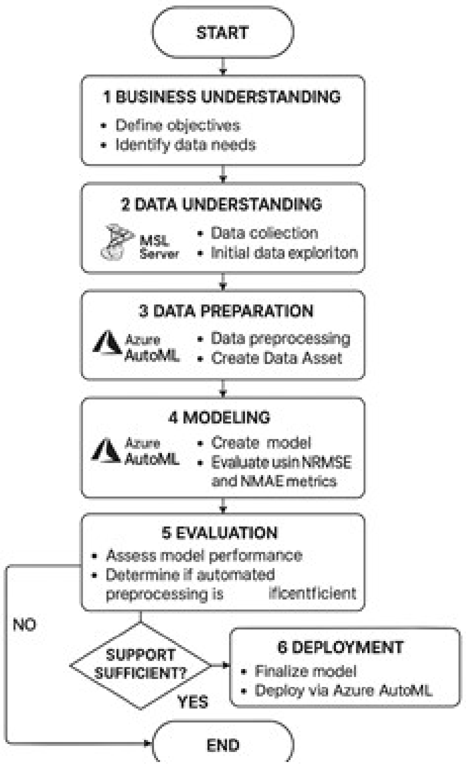

Figure 1. Research methodology using the CRISP-DM framework

4. RESULTS

4.1 Business Understanding

Company XYZ has difficulties in enhancing sales forecasting precision to facilitate strategic choices about production, inventory, and marketing, owing to the constraints of traditional spreadsheet-based approaches that depend on manual data handling and are susceptible to human error. The firm has used Azure AutoML to address inefficiencies, targeting semester forecast variations below 15% while improving supply chain efficiency and mitigating operational risks via automated model building. The implementation utilizes seamless integration within the Microsoft ecosystem and enables non-expert personnel through its no-code framework, strategically aligning technological adoption with operational goals to enhance competitiveness in Indonesia's flour market amid rising demand variability and industry challenges.

4.2 Data Understanding

At this step, data collection and preliminary analysis were performed, along with an assessment of the auto-preprocessing capabilities of Azure AutoML

4.2.1 Data Collection and Initial Analysis Data

This study employed two datasets: the first, a raw historical dataset covering five years (DSRawHistory_2019-2023.csv), necessitates data preprocessing and serves for initial analysis, training, and evaluation of Azure AutoML's automated preprocessing functions; the second, a refined dataset comprising actual monthly sales data for the first half of 2024 (DS-Actual-Mo-SQL-Jan24-Jun24.csv), is utilized to assess the performance of the developed models.

The preliminary study concentrated on semester sales data from Company XYZ over the period 2019 to 2023, quantified in metric tons, as shown in Table 1.

Table 1. Traditional and current sales forecasting statistics by semester

Year	1st Semester (January to June)				2nd Semester (July to December)			
	Product K		Product Y		Product K		Product Y	
	Forecast	Actual	Forecast	Actual	Forecast	Actual	Forecast	Actual
2019	1800	3060.28	10000	17185.8	1900	3152.53	13000	20783.6
2020	3200	1449.78	18000	18077.9	3800	2256.23	21500	20091.3
2021	1700	3831.88	18500	19898.6	2200	3690.58	21600	23261.9
2022	4000	1086.15	20000	7672.65	5000	2635.75	2900	13116.3
2023	2000	4343.08	7500	10546.9	2000	3370.35	12500	9687.63

The table 1 indicates considerable discrepancies between projected and actual sales for Products K and Y. In the first semester, significant discrepancies include Product K in 2022 (forecast: 4000 MT, actual: 1086.15 MT), leading to a -72.85% overestimation. In 2022, Product Y exhibited an overestimation of -61.64%, with a projected figure of 20,000 MT vs actual sales of 7,672.65 MT.

The second semester demonstrated significant underestimations, especially for Product Y in 2022, where the prediction was just 2900 MT, but actual sales amounted to 13116.28 MT (-352.28%). In 2023, Product K was overestimated by -68.52% (forecast: 2000 MT, actual: 3370.35 MT). These inconsistencies emphasize the shortcomings of traditional forecasting techniques and show the need for sophisticated strategies to mitigate risks related to inventory management. The irregular patterns across items and timeframes further demonstrate the difficulties in attaining precise forecasts using conventional methods.

4.2.2. Evaluation of the auto-preprocessing capabilities of Azure Auto-ML

An investigation was undertaken to assess the efficacy of Azure AutoML in facilitating automatic data preparation, concentrating on various essential aspects using a raw historical transaction dataset.

The results reveal that while Azure AutoML offers a degree of automation, its no-code interface exhibits considerable limits in comprehensively facilitating different preprocessing processes, hence requiring human involvement or the use of additional tools. The elimination of extraneous data is not inherently facilitated inside the no-code interface, necessitating user intervention or code-based solutions.

Likewise, the management of data duplication is not automated, necessitating users to manually eliminate duplicate entries prior to dataset upload. Conversely, variable selection is effectively facilitated by automated featurization during data asset construction, enabling the identification of relevant features based on their correlation with the target variable. Basic imputation techniques, including mean and median, are automatically provided for addressing missing data. Nevertheless, more intricate methods such as k-Nearest Neighbors (k-NN) need user involvement via the SDK. Format correction is restricted to the validation of data formats and does not provide automated repairs; discrepancies such as type mismatches or special characters must be addressed manually. Although outlier identification may be shown using tools such as box plots and histograms, the no-code interface does not facilitate direct management of outliers, necessitating the use of other tools or human coding. Azure AutoML has strong support for data aggregation, especially in time series analysis, including fundamental aggregation operations and rolling window computations. Data masking is not supported on the platform, necessitating the manual anonymization of sensitive data prior to upload.

In summary, the majority of data preparation functionalities in Azure AutoML are either partly supported or entirely unsupported in its no-code interface. Subsequently, data preparation was performed using MS SQL Server to guarantee the quality and preparedness of datasets prior to performing tests with AutoML

The findings of this investigation are shown in Table 2 below:

Table 2. Examination of Azure AutoML's capabilities for automatic data preprocessing

No.	Data Preprocessing Stage	Azure AutoML Support
1	Data Exclusion	No
2	Data Duplication	No
3	Variable Selection	Yes
4	Missing Values	Partial
5	Format Correction	No
6	Outliers	No
7	Data Aggregation	Yes
8	Data Masking	No

4.3 Data Preparation

At this juncture, data preparation was executed via MS SQL Server, and Data Asset development was carried out in Azure AutoML inside the Azure Machine Learning Studio environment. In this research, the temporal resolution was established as monthly to correspond with business requirements.

4.3.1 Data Preprocessing using MS SQL Server

Due to the constraints of automated data pretreatment in Azure AutoML, as previously analyzed, MS SQL Server was used as the principal instrument for executing all preprocessing activities prior to uploading datasets to Azure AutoML. This method utilizes the adaptability of SQL queries to address diverse preprocessing requirements [12]. Centralizing the whole pretreatment workflow on a singular platform guarantees data integrity, minimizes mistakes, and standardizes the procedures for further data processing. The preprocessing started with the importation of the raw transaction history information into the database as a table. Multiple preparation activities were performed with SQL queries. Initially, records devoid of enough trends across a five-year period (2,892 rows) were omitted from the dataset. Duplicate records were found (6 rows with 1 duplicate each, totaling 12 rows) were eliminated, resulting in only unique entries remaining. All fields were examined for missing values; while no null values were detected, 11 rows with zero or negative values in the TotalSales column were removed. Outlier identification was performed with two techniques: Interquartile Range (IQR) [8] and Z-Score [9]. Both methodologies were validated.

The dataset had a steady distribution devoid of notable outliers, indicating that no extra treatment was necessary for further analysis. The data formats and types were confirmed to be consistent, with MS SQL automatically modifying datatypes as required. The process of variable selection reduced the dataset to three essential columns: The DateShipment and ItemName were used to aggregate monthly total sales per product by adding TotalSales according to shipping dates and product names, hence creating a new column titled

SumTotalSales. Ultimately, data masking was executed by anonymizing product names to Product K and Product Y. After executing these procedures, the historical transaction processed dataset was exported as a CSV file titled 'DS-History-Mo-SQL-2019-2023.csv'. The final dataset had three columns: DateShipment (date) for shipping dates; ItemName (nvarchar) for anonymized product names; and SumTotalSales (decimal) for total sales in metric tons. This thorough pretreatment guaranteed that the dataset was pristine, uniform, and prepared for further analysis via Azure AutoML.

4.3.2 Data Asset Creation in Azure AutoML

At this point, the dataset was uploaded to Azure AutoML under the Azure Machine Learning Studio platform to facilitate machine learning experiments. The procedure included many stages, commencing with the upload of the preprocessed dataset 'DS-History-Mo-SQL-2019-2023.csv' as a data asset. The dataset, housed in an Azure Blob Storage repository, was prepared as a tabular CSV file with a semicolon (;) delimiter, UTF-8 encoding, and uniform column headers. Upon upload, Azure AutoML autonomously created a data preview and identified data types for each column. Nonetheless, schema identification problems emerged, shown by the column SumTotalSales being erroneously classified as a string owing to Indonesian decimal formatting (commas as delimiters). This required manual adjustment to change the column type to decimal (dot format) for alignment with Azure's global standards. Upon evaluating the dataset's structure, exploratory analysis verified that the essential columns (DateShipment, ItemName, and SumTotalSales) were appropriately structured and prepared for machine learning. The dataset met all criteria for time series forecast experiments. Following this preparation, the dataset was deemed suitable for indicates higher model accuracy and better fit to the observed data.

Subsequent to the upload of the historical dataset, the procedure proceeded with the upload of the current six-month dataset (DS-real-Mo-SQL-Jan24-Jun24.csv) as a data asset, which would subsequently serve as a benchmark for juxtaposing real data with the model's predictions.

4.4 Modelling

Conventional machine learning methodologies need data scientists to pre-establish models according to domain-specific criteria, requiring substantial knowledge and effort commitment. Conversely, Azure AutoML automates the processes of model selection, hyperparameter optimization, and feature engineering according to Task Type and Primary Metric parameters. This research used Azure AutoML's 'Train Automatically' feature for time series forecasting using the data asset 'DS-History-Mo-SQL-2019-2023.csv', capitalizing on its no-code functionalities to enhance analytical processes. Performance assessment employs Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) to measure prediction accuracy between real and predicted numbers [6]. To resolve scale dependence issues that hinder cross-dataset comparisons, Azure AutoML employs normalized metrics, including Normalized Mean Absolute Error (NMAE) and Normalized Root Mean Squared Error (NRMSE), facilitating standardized evaluation across datasets with differing magnitude ranges [13]. NMAE is a statistic used to assess the accuracy of prediction models by normalizing the MAE in relation to a selected scale, such as the range, mean, or sum of actual data. Reduced NMAE values indicate superior model performance. NRMSE is a statistic used to assess the accuracy of prediction models by normalizing the RMSE in relation to the range, mean, or standard deviation of observed data. A diminished NRMSE indicates higher model accuracy and better fit to the observed data. Let n represent the total number of samples, yi denote the actual value, and $\hat{y}i$ signify the predicted value, while ymax indicates the greatest value of real observations and ymin represents the lowest value of actual observations.Computational limitations were established at 150 trials, 4 concurrent trials, 2 nodes, a 720-minute experiment timeout, and a 60-minute iteration timeout.K-Fold Cross Validation with five folds was used because of the dataset's small (fewer than 20,000 rows).A CPU-based virtual machine (2 cores, 14GB RAM, 100GB storage) was used, and the task was submitted for automated model training and selection.

Table 3. Procedures and criteria for modeling

No.	Steps	Values
1	Training Method	Train Automatically
2	Job Name	NMAE-Horizon-Mo
		NRMSE-Horizon-Mo
3	Task Type	Time Series Forecasting
	Data	DS-History-Mo-SQL-2019-2023.csv
4	Task Settings	
	Target Column	SumTotalSales

	Time column	DateShipment
	Time series identifier	ItemName
	Frequency	Month
	Forecast Horizon	6
	Primary Metric	NMAE and NRMSE
	Featurization	Auto Enabled
	Limit Max Trials	150
	Max Concurrent Trials	4
	Max Nodes	2
	Experiment timeouts	720 min.
	Iteration timeouts	60 min.
	Validation type	K-Fold Cross Validation
	Number of cross validations	5
5	Compute type	Cluster
	VM Type	CPU - Dedicated
	VM Size	2 cores, 14Gb Ram, 100Gb Storage

4.5 Evaluation

At this juncture, the assessment of the optimal model and its efficacy was conducted based on modeling outcomes, alongside model testing, to verify that the model provides dependable performance prior to its deployment in a production setting

4.5.1 Best Model

The Normalized Root Mean Squared Error (NRMSE) and Normalized Mean Absolute Error (NMAE) measures consistently selected the VotingEnsemble as the superior model. The VotingEnsemble model amalgamates predictions from separate models, such as MinMaxScaler-ExtraTreesRegressor, ExponentialSmoothing, and Naïve, with a soft voting methodology [14]. Each model produces predictions with different methodologies:MinMaxScaler - ExtraTreesRegressor normalizes data and identifies intricate patterns; ExponentialSmoothing prioritizes recent data to address trends and seasonality; Naïve depends only on the most recent historical value, disregarding trends and seasonality. The final prediction ($PFinal$) is computed as a weighted average of the various forecasts.

Models with greater weights have a more substantial influence on the final forecast, so enabling the ensemble to capitalize on the advantages of superior models while alleviating their deficiencies. This methodology yields more precise and consistent forecasts than depending on an individual model.

4.5.2 Best Metric

The NRMSE reached 0.26622, signifying that prediction mistakes constituted around 26.622% of the range or mean of actual data, while the NMAE obtained 0.21278, indicating that absolute prediction errors averaged 21.278% of the range or mean. The VotingEnsemble model has superior accuracy when assessed using NMAE compared to NRMSE. The model with NMAE was chosen as the ideal candidate for deployment.

4.5.3 Model Performance Ranking

Azure AutoML performs automatic model selection via 150 trial iterations, according to the maximum trials configuration, which includes individual models and normalized data-model combinations. Table 4 displays the top 10 models listed in decreasing order according to the key NMAE metric value.

Table 4. Evaluation of the top 10 models based on the NMAE metric

No.	Model	NMAE	MAE
1	MaxAbsScaler, DecisionTree	0.29749	640.36
2	SeasonalAverage	0.29521	631.01
3	MinMaxScaler, GradientBoosting	0.28846	565.98
4	AutoArima	0.26224	584.15
5	RobustScaler, ExtremeRandomTrees	0.25374	546.67
6	StandardScalerWrapper, XGBoostRegressor	0.25354	519.31
7	StandardScalerWrapper, RandomForest	0.24537	514.7
8	Naïve	0.22367	439.82
9	ExponentialSmoothing	0.21829	438.27
10	VotingEnsemble	**0.21278**	**424.35**

4.5.4 Validating the best model's predictive reliability with actual data

Azure AutoML has a Model Testing tool intended to assess the dependability and accuracy of machine learning models produced during training. This feature enables users to evaluate the optimal model with a distinct test dataset, yielding essential outputs like evaluation metrics and prediction files. Model testing is essential for confirming predictions in practical contexts, facilitating informed decision-making and bolstering trust in the model's efficacy. This study employs the Model Testing feature, utilizing the 'DS-Actual-Mo-SQL-Jan24-Jun24.csv' data asset as a benchmark to accomplish two primary objectives: (1) to regenerate the NMAE metric, and (2) to generate the 'prediction.csv' file, which is subsequently used to assess the performance of the optimal model (VotingEnsemble) by comparing its predictions with actual data for the same period. From January to June 2024, guaranteeing dependability prior to implementation and offering significant insights into predictive precision. NMAE Metric Performance in Model Evaluation Upon recalculating evaluation measures during testing, the model attained an NMAE of 0.12118, reflecting an improvement relative to the training NMAE of 0.21278. This signifies that the model generalizes.

The model has robust predictive performance with fresh data, making it appropriate for implementation. Comparison of Predicted Outcomes and Actual Results from Model Evaluation Table 5 below presents the projected and actual values for the period from January 2024 to June 2024, derived from the output file 'prediction.csv,' and will be subjected to additional analysis.

Table 5. Forecasted vs actual outcomes for the first semester

Products	Jan-24		Feb-24		Mar-24	
	Forecast	Actual	Forecast	Actual	Forecast	Actual
Product K	464.38	448.85	458.05	346.65	461	413.45
Product Y	1447.41	1918.63	1438.42	1513.9	1430.75	1125.97

Products	Apr-24		May-24		Jun-24	
	Forecast	Actual	Forecast	Actual	Forecast	Actual
Product K	468.64	489.95	463.72	496.05	458.29	367.05
Product Y	1420.61	1098.03	1428.99	1691.32	1425.43	1529.68

The analysis was performed in two stages to juxtapose its predictions with real facts. A monthly deviation analysis was conducted, followed by a relative deviation average analysis over the six-month first semester period. Both analyses offer insights into predictive accuracy, with smaller deviations signifying greater accuracy.

Monthly Deviation Analysis (2.1): This study aims to quantify the proportion of variance or deviation between projected and actual data weekly, using the formula for Relative Deviation proportion (RDP).

Table 6. Monthly relative deviation forecasts against actual outcomes

2024	Deviation (%)	
	Product K	Product Y
Januari	3.46	−24.56
Februari	32.14	−4.99
Maret	11.48	27.07
April	−4.35	29.38
Mei	−6.52	−15.51
Juni	24.86	−6.82

(2.2) Analysis of average relative deviation over one semester: this computes the mean of relative deviations over a defined timeframe, namely six months (one semester). The Relative Deviation Average (RDA) for one semester may be computed using a designated mathematical procedure, as shown by the monthly relative deviation percentages in Table 5.

4.5.5 Final Analysis of Predictions vs Actuals

For Product K, the average relative deviation (RDA) is 10.18% over six months, indicating robust performance that closely corresponds with the overall model performance (NMAE 0.21278). The peak deviation occurred in February 2024 at 32.14%, where predictions surpassed actual sales, while lesser deviations in April (-4.35%) and May (-6.52%) reflect precise predictions, demonstrating the model's efficacy in capturing the sales patterns of this product.

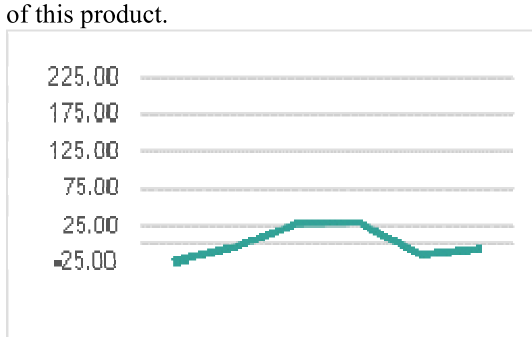

Figure 3. Monthly variance chart for product Y

(2) For Product Y, the model exhibits exceptional performance with an average relative deviation (RDA) of merely 0.76% over six months, accurately aligning with actual data overall, notwithstanding the variances in March 2024 (27.07% deviation, predictions exceeding actuals) and April 2024 (29.38% deviation, predictions falling short of actuals), ultimately yielding satisfactory outcomes for this product.

(3) Comparison of RDA between Conventional Method and Azure AutoML: Table 8 below presents a comparison between the Conventional approach and Azure AutoML using the identical formulae.:

Table 8. Comparison of RDA: Conventional Method against Azure AutoML

Year	RDA (%)		Method
	Product K	Product Y	
2019	-40	-41.81	Conventional
2020	120.73	-0.43	Conventional
2021	-55.63	-7.56	Conventional
2022	-72.85	-61.64	Conventional
2023	-53.91	-28.84	Conventional
2024	**10.18**	**0.76**	**Azure AutoML**

The Azure AutoML model attained markedly reduced Relative Deviation Averages (RDA) for Product K (10.18%) and Product Y (0.76%) in contrast to traditional methodologies, which historically displayed significant volatility (e.g., Product K: −72.85% to 120.73%; Product Y: −61.64%), thereby illustrating its enhanced capacity to minimize forecasting inaccuracies and stabilize predictions.

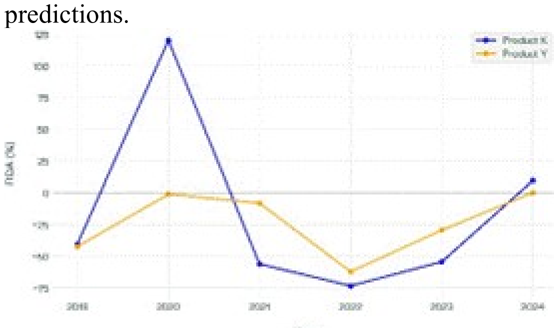

Figure 4. Comparison of Suggested Nutritional Intakes Comparison of Traditional Method with Azure AutoML Chart

4.6 Deployment

The deployment procedure in Azure AutoML entails registering the optimal model (VotingEnsemble) within the Azure Machine Learning workspace, establishing an online endpoint with key-based authentication for secure access, and deploying the model to the endpoint utilizing Azure Container Instance (ACI) for real-time inference.

5. DISCUSSION

This research successfully fulfilled its key goals by using Azure AutoML, which significantly enhanced sales forecasting accuracy, resolved preprocessing difficulties, and enabled non-expert users via its no-code interface. The VotingEnsemble model was the most effective solution, attaining a Normalized Mean Absolute Error (NMAE) of 0.21278 in training and improving to 0.12118 in testing. For Product K, the model attained a six-month Relative Deviation Average (RDA) of 10.18%, but Product Y demonstrated remarkable precision with an RDA of 0.76%, much surpassing traditional approaches that revealed considerable variability (e.g., Product K: −72.85% to 120.73%; Product Y: −61.64%).
Nonetheless, Azure AutoML's inherent preprocessing functionalities were inadequate for intricate tasks like data exclusion and outlier management, necessitating user intervention via MS SQL Server. This hybrid methodology guaranteed data integrity and compatibility, successfully tackling the preprocessing problems outlined in Objective 2. The no-code interface allowed non-expert users to deploy models within Company XYZ’s Microsoft ecosystem; however, a fundamental understanding of statistics and data preprocessing techniques, including outlier detection via Z-Score and IQR, along with proficiency in MS SQL Server (Transact-SQL), was essential for effective implementation [17][18].

6. CONCLUSION

This research confirms the efficacy of Azure AutoML in enhancing sales forecasting precision for stable-demand items, with errors below 15% for items K and Y, while markedly surpassing traditional methodologies. The VotingEnsemble model exhibited strong predictive accuracy, closely matching predictions with actual sales data, thereby achieving the main goal of improving forecast dependability.
The amalgamation of the CRISP-DM approach, manual preprocessing via MS SQL Server, and Azure AutoML’s no-code architecture yielded a scalable solution for Company XYZ’s Microsoft-centric environment.
The no-code method has democratized AI adoption for non-expert users; yet, successful application requires a fundamental understanding of data preparation techniques and statistical principles.
Future research should concentrate on refining AutoML's automation pipeline to mitigate preprocessing constraints and investigating dynamic variable integration to enhance forecasting precision across various product portfolios.

7. ACKNOWLEDGEMENTS

No potential conflict of interest was reported by the authors. No funding is sourced for this research. Author contributorship: Fikri Salam - conceptualization, formal analysis, investigation, methodology, resources, visualization, writing and editing; Tanty Oktavia - conceptualization, formal

analysis, project administration, valida-tion and review.

Data openness and transparency. To facilitate further research, we have made the data used in their study publicly available. Data supporting this study is available from ze-nodo at https://doi.org/10.5281/zenodo.15295353

REFERENCES

[1] Xu, S. (2023). Comparison of Sales Prediction in Conventional Insights and Machine Learning Perspective. Journal of Business and Management, 38, 1681–1689.

[2] Kuehl, N., Schemmer, M., Goutier, M., & Satzger, G. (2022). Artificial intelligence and machine learning. EM, 32(4), 2235–2244.

[3] He, X., Zhao, K., & Chu, X. (2021). AutoML: A survey of the state-of-the-art. Knowledge-based Systems, 212, 106622.

[4] Paranjape, A., Katta, P., & Ohlenforst, M. (2022). Automated data preprocessing for machine learning based analyses. COLLA 2022: The Twelfth International Conference on Advanced Collaborative Networks, Systems and Applications.

[5] Guyon, I., Sun-Hosoya, L., Boullé, M., Escalante, H. J., Escalera, S., Liu, Z., Jajetic, D., Ray, B., Saeed, M., Sebag, M., Statnikov, A, Tu, W., & Viegas, E. (2019). Analysis of the AutoML Challenge Series 2015–2018. In The Springer series on challenges in machine learning (pp. 177–219).

[6] Soto-Ferrari, M., Chams-Anturi, O., & Escorcia-Caballero, J. P. (2020). A time-series forecasting performance comparison for neural networks with state space and ARIMA models. Proceedings of the 5th North American International Conference on Industrial Engineering and Operations Management, 159–162.

[7] Rohaan, D., Topan, E., & Groothuis-Oudshoorn, C. (2022). Using supervised machine learning for B2B sales forecasting: A case study of spare parts sales forecasting at an after-sales service provider. Expert Systems With Applications, 188, 115925.

[8] Pavlyshenko, B. M. (2019). Machine-Learning Models for Sales time Series Forecasting. Data, 4(1), 15.

[9] Westergaard, G., Erden, U., Mateo, O.A., Lampo, S.M., Akinci, T.C., & Topsakal, O. (2024). Time Series Forecasting Utilizing Automated Machine Learning (AutoML): A Comparative Analysis Study on Diverse Datasets. Information, 15(1), 39.

[10] Tarallo, E., Akabane, G. K., Shimabukuro, C. I., Mello, J. a. V. B., & Amancio, D. (2019b). Machine Learning in Predicting Demand for Fast-Moving Consumer Goods: An Exploratory research. IFAC-PapersOnLine, 52(13), 737–742.

[11] Wirth, R., & Hipp, J. (2000). CRISP-DM: Towards a Standard Process Model for Data Mining. Dalam Proceedings of the 4th International Conference on Practical Applications of Knowledge Discovery and Data Mining (p. 29-39).

[12] Schröer, C., Kruse, F., & Gómez, J. M. (2021). A Systematic Literature Review on Applying CRISP-DM Process model. Procedia Computer Science, 181, 526–534.

[13] Roy, S. (2024). Net Sales Prediction – Azure Machine Learning. Master's Thesis, Haaga-Helia University of Applied Sciences

[14] Jabbar, H. G. (2024). Advanced Threat Detection Using Soft and Hard Voting Techniques in Ensemble Learning. Journal of Robotics and Control (JRC), 5(3), 1108–1111.

[15] Brownlee, J. (2020). How to Develop a Weighted Average Ensemble for Deep Learning Neural Networks. Machine Learning Mastery.

[16] Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. International Journal of Forecasting, 22(4), 679–688.

[17] T. Oktavia, "Implementing Data Warehouse As A Foundation For Decision Support System (Perspective: Technical And Nontechnical Factors)," J Theor Appl Inf Technol, vol. 60, no. 3, 2014.

[18] F. Siska and T. Oktavia, "Predictive Analysis of Hypertensive Heart Disease Using a Machine Learning Approach," in 11th International Conference on ICT for Smart Society: Integrating Data and Artificial Intelligence for a Resilient and Sustainable Future Living, ICISS 2024 - Proceeding, 2024. doi: 10.1109/ICISS62896.2024.10751517.