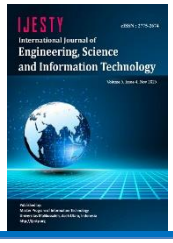
International Journal of Engineering, Science and Information Technology
Volume 5, No. 4 (2025) pp. 64-74
eISSN: 2775-2674
Website: http://ijesty.org/index.php/ijesty
DOI: https://doi.org/10.52088/ijesty.v5i4.979
Research Paper, Short Communication, Review, Technical Paper



Comparison of LSTM and TCN Models for Customer Churn Prediction Based on Sentiment and Transaction Data

Made Bayu Brahmanda Dharmasaguna*, Astari Retnowardhani

Department of Information Systems Management, Bina Nusantara University, Jakarta, Indonesia

**Corresponding author Email: made.dharmasaguna@binus.ac.id*

The manuscript was received on 21 February 2025, revised on 1 May 2025, and accepted on 22 August 2025, date of publication 3 November 2025

Abstract

This study investigates the combined use of customer review sentiment analysis and transaction history to predict customer churn on the Balimall Market e-commerce platform. The dataset includes 41,519 reviews labeled with positive and negative sentiments and 48 transaction samples labeled as churn or non-churn based on RFM method. Two deep learning models, Long Short-Term Memory (LSTM) and Temporal Convolutional Network (TCN), are applied in parallel for each analysis path. Data pre-processing includes filtering, cleaning, tokenizing, normalization, sentiment labeling, as well as feature engineering and churn labeling. Evaluation using accuracy, precision, recall, F1-score, and confusion matrix metrics shows that TCN excels with 91.55% accuracy on sentiment analysis and 91.67% on churn prediction, while LSTM achieves 86.35% and 86.67% respectively. Segment analysis shows that 47.30 % of users express negative sentiment yet remain active, 51.69 % express positive sentiment and remain active , 0.54 % express negative sentiment and churn, and 0.48 % express positive sentiment and churn. This finding demonstrates that negative sentiment alone does not necessarily lead to churn; instead, the greatest churn risk arises in negative sentiment churners and positive sentiment churners. Expert validation confirmed the reliability of both models, with the recommendation of using a hybrid to combine the advantages of each architecture. The results of this study are expected to help Baliyoni Group design a more targeted customer retention strategy and improve customer satisfaction by examining these segment conditions.

Keywords: *Churn, LSTM, TCN, E-commerce, Sentiment Analysis.*

1. Introduction

The development of e-commerce in Indonesia in recent years has shown significant growth. The increase in the value of digital transactions every year, the expansion of internet access which now reaches more than 70% of the population, and the shift in consumer behavior towards online platforms are the main factors in this trend [1]. However, customer churn remains a serious challenge that can reduce revenue and threaten business sustainability [2][3]. This research aims to provide a comprehensive understanding of customer behavior by combining sentiment analysis of user comments and transaction history based on the RFM (Recency, Frequency, Monetary) method [4][5]. This approach allows the identification of customer segments that despite negative feedback remain active, as well as those who were initially positive but later stopped transacting [6][7][8]. To improve the accuracy of churn detection, this study applies two deep learning techniques, Long Short-Term Memory (LSTM) and Temporal Convolutional Network (TCN), and compares their performance. The LSTM method is chosen for its ability to capture long-term context in text reviews, while TCN is preferred for its efficiency in recognizing temporal patterns in transaction data [9][10]. The analysis stages include text data pre-processing, RFM feature extraction, model training, and evaluation using accuracy, precision, recall, and F1-score metrics [11][12]. With the results of this analysis, Baliyoni Group (Balimall and Tokodaring) can formulate more appropriate customer retention strategies, such as service improvements based on negative feedback and more personalized marketing efforts. The key question in this research is which group has a greater impact on customer retention, users with negative sentiment who remain active, or users with positive sentiment who then stop transacting. The answer to this question is expected to help Baliyoni Group prioritize service development.

2. Literature Review

2.1. Sentiment Analysis with Deep Learning

Deep learning has emerged as the leading approach for sentiment analysis, effectively handling the complexities of natural language in e-commerce reviews. Hybrid architectures such as CNN-LSTM leverage convolutional layers to extract local word patterns, followed by LSTM units to capture long-range contextual dependencies in text. Implementing this hybrid design has been shown to improve sentiment classification accuracy by over four percentage points compared to traditional machine-learning methods [13][14]. Additionally, combining an attention mechanism with LSTM and Word2Vec embeddings effectively focuses processing on key sentiment terms, yielding an average F1-score increase of 3.5 points on multilingual review datasets [15][16]. More recently, Temporal Convolutional Networks (TCNs) have been adopted for short-text sentiment analysis. By processing sequential data in parallel using dilated convolutional filters, TCNs avoid vanishing-gradient issues and demonstrate stable training, achieving F_1-scores of up to 90 % on brief reviews [17]. Ensemble models that merge LSTM and TCN outputs have also been evaluated, mitigating each architecture's individual weaknesses and delivering more consistent, superior performance across multiple evaluation metrics [18][19].

2.2. Text Mining and Text Processing

Recent studies confirm that text mining and thorough text preprocessing are foundational to the development of effective sentiment-analysis and churn-prediction models in e-commerce. Comprehensive preprocessing begins with text normalization, which standardizes nonstandard spellings and removes emoticons to ensure data consistency [20]–[22]. Filtering techniques eliminate non-informative tokens, including stopwords and special characters, thereby reducing noise that can degrade model performance [23]. Further cleaning removes URLs, numbers, symbols, and HTML markup, which enhances corpus quality and accelerates training convergence [24]. Tokenization then splits text into analytical units (words or phrases), playing a key role in frequency and co-occurrence-based feature extraction [25]. Finally, sentiment and churn labels are assigned manually or semi-automatically to establish ground truth, enabling deep models to learn relevant patterns [26]. An end-to-end implementation of these text-mining stages is crucial for achieving stable training and high accuracy with both LSTM and TCN models in the Balimall Market domain.

2.3. Feature Engineering for Churn Prediction

Feature engineering is a critical step in preparing transaction data for churn-prediction models. Early work applied the RFM framework, namely recency, frequency, and monetary value, as basic indicators of churn risk [27]. Later research added user-engagement metrics such as session counts per period and visit duration derived from clickstream analysis to capture dynamic behavior patterns [28]. Time-to-event weighting was introduced to emphasize recent transactions more heavily, enabling models to detect declines in customer activity more sensitively [29]. Advanced transactional features such as average order value, visit-to-purchase conversion ratio, and count of unique products purchased also enrich the feature set and sharpen predictive accuracy [30]. To capture ongoing temporal trends, rolling-window metrics and churn-propensity scores aggregate historical data over defined intervals so that deep models learn both short-term and long-term patterns [31]. Numerical transformations, categorical encoding, and feature normalization ensure consistent feature scales and maximize the performance of LSTM and TCN architectures in churn prediction.

2.4. Aplication of LSTM and TCN Models in Churn and Sentiment Analysis

The application of LSTM and TCN models for sentiment analysis and churn prediction in e-commerce has been extensively explored in the literature. Initial implementations of LSTM demonstrated its ability to learn long-range dependencies in customer-review sequences, although it remains susceptible to vanishing gradients on very long texts [32]. In contrast, TCNs process sequential data in parallel and offer stable training dynamics with faster convergence. Comparative studies on historical transaction data show that LSTM excels at modeling prolonged purchase sequences, while TCN outperforms on datasets with sporadic transactions by leveraging dilated convolutions to detect temporal patterns at multiple scales [33]. Hybrid approaches that combine text embeddings from LSTM with convolutional outputs from TCN have delivered synergies that improve F1-scores by 2 to 3 points in various experiments. Recent comparative work has also optimized architectures and hyperparameters for both models: LSTM uses dropout and recurrent-dropout to prevent overfitting, whereas TCN employs residual blocks and L2 regularization for stability. Results indicate that TCN is generally more robust to noise in transaction and review data, while LSTM offers richer contextual embeddings for long-range dependencies [34][35]. Building on these findings, this study adapts and compares LSTM and TCN models on Balimall Market data using binary sentiment labels (positive and negative) and churn status to determine the most suitable architecture for the local e-commerce context.

2.5. Integration of Sentiment Analysis and Transaction History

Two recent studies have combined customer review sentiment analysis and transaction history within deep learning frameworks to predict customer churn. The first study introduced a fusion method in which text-embedding vectors from sentiment analysis are concatenated directly with RFM transaction features as inputs to a multilayer perceptron. This approach improved accuracy by 5 % compared to models using only one data type [36]. The second study proposed a hybrid method based on graph neural networks to model interrelations between sentiment and transactional features. In this method, sentiment features form text nodes, RFM metrics become transaction nodes, and connections are optimized through message passing. This approach yielded an average F1-score increase of 4.2 % and greater performance stability under shifting data distributions [37]. Both studies highlight the effectiveness of architecturally merging qualitative and quantitative data in a single model. These findings motivate the present research to explore similar integrative LSTM and TCN architectures for the Balimall Market domain.

3. Method

This chapter presents our system overview, the data preprocessing stages, feature engineering, through to model design and training and performance evaluation.

3.1. System Overview

The research system is designed to process both customer review data and transaction data from e-commerce. The workflow begins with data collection (Get Data), obtaining 41,519 records, then enters the preprocessing phase, which is divided into two streams, the sentiment analysis stream and the churn prediction (transaction history) stream. In the sentiment analysis stream, text data are processed through filtering, cleaning, tokenizing, normalization, and labeling stages so they are ready for deep learning models. In the churn prediction stream, transaction data are cleaned, engineered, and labeled to produce numerical features. After preprocessing, the sentiment analysis stream yields 41,519 review samples, split into training data 80% (33,215), validation data 10%(4,152), and test data 10% (4,152). The churn prediction stream yields 444 transaction samples, split into training data 80% (356), validation data 10% (44), and test data 10% (44). Next, in the Model Training stage, the LSTM and TCN models are trained on these data. The final stage is Evaluation and Results, where performance is measured and validated by experts.

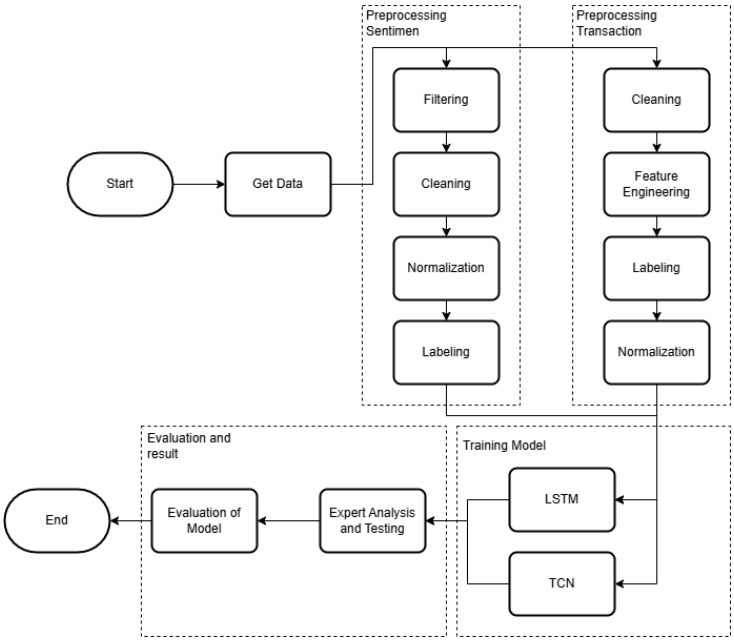


Fig 1. System Flow Diagram

In this study, model optimization was conducted through refinement of the network architecture and the application of regularization techniques to maintain balanced learning between the sentiment preprocessing stream and the churn preprocessing stream without sacrificing important information in either pipeline. Furthermore, the use of a softmax activation function together with categorical cross-entropy loss effectively controls the predicted probability distribution, reducing bias toward the majority class. This approach enables the model to accurately recognize sentiment patterns from reviews and numerical transaction features without requiring additional techniques such as oversampling or reweighting.

3.2. Filtering

Filtering starts by examining the reviews obtained from the e-commerce platform and discarding any content that does not aid the analysis. In this phase all text is converted to lowercase for consistency, digits and characters that are not letters are removed, URLs without informative value are deleted, user mentions and external links are stripped out, and hashtags that lack analytical relevance are discarded.

1. Initial text: "*kualitas produk bagus,, mantap*"
2. Filtered result: "*kualitas produk bagus mantap*"

3.3. Cleaning

The cleaning stage aims to refine the filtered output by removing unnecessary text elements and simplifying word forms. This process includes removing stopwords, such as "*dan*," "*yang*," or "*atau*," that do not contribute significantly to the analysis, and applying stemming to reduce words to their base form, for example changing "*pesanan*" to "*pesan*" or "*makanan*" to "*makan*." These steps help reduce noise in the data and ensure a more informative and consistent text corpus structure. In the churn-prediction preprocessing stream, the cleaning stage also involves dropping any rows with missing values in critical RFM analysis columns, such as user_id, order_id, created_at, and total_price. This step ensures that only complete and valid transaction data are used for calculating RFM metrics and labeling churn.

3.4. Tokenizing

After the cleaning stage, tokenizing splits text into small units called tokens, which may be single words or short phrases. This process allows the model to process each token separately and prepares the data for feature extraction and deeper analysis. For example, the sentence "*pelanggan memberikan ulasan sangat memuaskan dan pengiriman cepat*" is split into the tokens "*pelanggan*", "*memberikan*", "*ulasan*", "*sangat*", "*memuaskan*", "*dan*", "*pengiriman*", "*cepat*". This approach is crucial for tasks such as word-frequency computation or syntactic evaluation, as the model can learn patterns at the token level before proceeding to further analysis.

3.5. Feature Engineering

At the feature engineering stage, transaction data are aggregated at the user level by computing several key metrics for RFM analysis. First, transaction_count represents the number of unique orders per user and reflects transaction frequency. Second, visit_frequency counts the total number of visits based on transaction records. Third, last_transaction denotes the date of each user's most recent transaction and is used to calculate recency_days, which is the difference in days between the analysis date and the date of the most recent transaction. Fourth, monetary_value sums the total purchase amount per user to measure monetary contribution. The calculation of recency_days is performed by subtracting the date of the most recent transaction from the analysis date. This process produces structured numerical features that are ready for normalization and for use as inputs to the churn prediction model.

3.6. Normalization

During normalization, text is standardized according to the Great Dictionary of the Indonesian Language (KBBI). This process includes expanding abbreviations, such as changing "*dgn*" to "*dengan*" or "*bgtu*" to "*begitu*", and unifying letter formats for consistency. This stage ensures text-data consistency and facilitates subsequent analysis. In the churn-prediction preprocessing stream, RFM features such as transaction_count, visit_frequency, monetary_value, and recency_days are normalized with MinMaxScaler to the range 0 to 1. This step guarantees that all numerical features share a uniform scale and are ready for model training.

3.7. Labelling

At the labeling stage, each text unit is manually evaluated by the evaluator team to determine its sentiment category, either positive or negative. The evaluators read and analyze the content of each customer comment to assign the label that best fits the context. Example:

1. Sample text: "*barang sesuai pesan kualitas produk bagus*"
2. Assigned label: positive

In parallel, in the churn-prediction stream, labeling is performed automatically based on the defined churn criteria. Customers who have not made a transaction within a specified period (for example, more than 180 days) and fall into the bottom 40 percentile for visit frequency and monetary value are assigned the label "1" (churn). In contrast, customers who remain active during that period are assigned the label "0" (non-churn).

3.8. Review and Transaction Data Before and After Preprocessing

This sub-section describes the selection of review-text and historical-transaction variables, the preprocessing steps for each (text cleaning, normalization, tokenization, and sentiment labeling for reviews; format standardization, RFM metric aggregation, and churn labeling for transactions), and presents the before-and-after preprocessing results in Tables I–IV.

Table 1. Review Data Before Preprocessing

user_id	user_name	review
58	Ni Luh Candra Darmayanti	*Kualitas buruk, barang rusak saat diterima.*
40	I Putu Sugi Almantara	*barang sesuai pesanan,, kualitas produk bagus*
82	anak agung putu mahendra putra	*Terimakasih, sesuai dg pesanan.. mantap..*
71	Ni Putu Diah Suani Arsini	*Pelayanan lambat, tidak profesional.*
61	Swastika Widya Mahasena	*Packing bagus, pengiriman cepat, top pokoknya. Mantap.*

Table 1 presents a sample of the original review data before preprocessing. The user_id and user_name columns identify each user, while the review column contains raw text that still exhibits spelling variations, excessive punctuation, and informal expressions. This view highlights the noise present in the text data that must be cleaned prior to analysis.

Table 2. Review Data After Preprocessing

user_id	user_name	review	label
58	Ni Luh Candra Darmayanti	*Kualitas buruk barang rusak saat diterima.*	Negatif
40	I Putu Sugi Almantara	*barang sesuai pesan kualitas produk bagus*	Positif
82	anak agung putu mahendra putra	*terimakasih sesuai dengan pesan mantap*	Positif
71	Ni Putu Diah Suani Arsini	*layanan lambat tidak profesional*	Negatif
61	Swastika Widya Mahasena	*bagus kirim cepat top pokok mantap*	Positif

Table 2 presents a subset of the review data after text preprocessing, which includes removal of duplicate punctuation, conversion to lowercase, and normalization of informal expressions (for example, "*Terimakasih, sesuai dg pesanan.. mantap..*" becomes "*terimakasih sesuai dengan pesan mantap*"). The cleaned reviews now exhibit consistent phrasing and are ready for analysis. Each entry has been assigned a sentiment label (Negative or Positive) so that the deep learning model can learn the characteristic word patterns associated with each sentiment class.

Table 3. Transaction Data Before Preprocessing

user_id	order_id	created_at	total_price
6	58	2022-09-20 14:21:47	1,275,000

user_id	order_id	created_at	total_price
8	34	2022-10-02 12:27:22	175,000
9	6	2021-09-29 10:53:02	14,200,000
10	103	2022-10-17 14:50:44	2,500,000
18	96	2022-10-21 09:12:48	13,000

Table 3 displays a snippet of the raw transaction data before feature engineering. Each row lists user_id, order_id, the transaction timestamp (created_at), and the purchase amount (total_price) in its original currency and datetime format. This presentation underlines the formatting inconsistencies that must be standardized in the subsequent stage.

Table 4. Transaction Data After Preprocessing

user_id	transaction_count	visit_frequency	last_transaction	monetary_value	recency_days	churn
6	0,002427184466	0,001580715274	2021-03-17 12:27:56	0,00240404614	0,8197747184	0
8	0	0	2020-07-22 13:53:15	0,0000001931589103	0,9687108886	1
9	0,0007281553398	0,0003951788184	2022-03-11 10:04:07	0,0002814132165	0,5951188986	0
10	0	0	2020-08-11 18:19:38	0,00000747524983	0,9555694618	1
18	0,003155339806	0,002766251729	2021-03-23 08:44:41	0,0007212862766	0,816020025	0

Table 4 presents the feature-engineered transaction metrics after min–max normalization. The transaction_count, visit_frequency, monetary_value, and recency_days columns have been scaled to the [0,1] range to ensure that all numerical inputs contribute equally during model training. The last_transaction column retains the original timestamp for audit and potential time-based analyses. Finally, the churn flag (0 = non-churn, 1 = churn) is derived from our RFM criteria. These normalized features and binary labels are now ready for direct input into the churn-prediction model.

3.8. LSTM

In this study, the LSTM model is applied in parallel to both the sentiment analysis stream and the churn prediction stream. The model accepts sequences of text tokens and numerical features with matching input dimensions. Each token is converted into a 128-dimensional vector via an embedding layer. Two sequential LSTM layers, with 128 and 64 units respectively and a dropout rate of 0.2, capture sequential patterns in the data. The resulting output is passed through a dense layer of 64 neurons with ReLU activation to reduce dimensionality, and then mapped into two classes, positive and negative sentiment, via an output layer with softmax activation. In the sentiment analysis stream, the model is trained for 100 epochs with a batch size of 32. In the churn prediction stream, the batch size is 16. The training process uses the Adam optimizer with a learning rate of 0.001. The overall model structure is shown in Fig. 2.

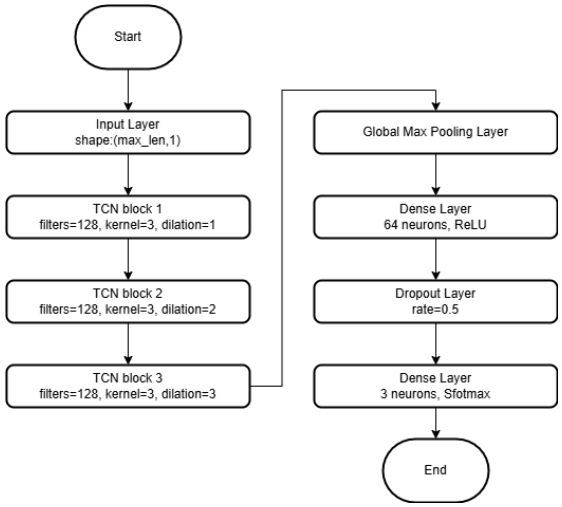


Fig 2. The structure of LSTM model implementation

3.9. TCN

The TCN model is applied in parallel to both the sentiment analysis stream and the churn prediction stream. The input data consist of sequences of text tokens and numerical features with shape (max_len, 1), which are processed through four consecutive TCN blocks, each with 128 filters, a kernel size of 3, and dilation rates of 1, 2, 4, and 8. The outputs of these blocks are aggregated by a Global Max Pooling layer to reduce dimensionality. The pooled output is then passed to a dense layer of 64 neurons with ReLU activation, followed

by a dropout rate of 0.3 to minimize the risk of overfitting. The output layer consists of two neurons with softmax activation to classify sentiment as positive or negative. The model is trained using the Adam optimizer with a learning rate of 0.001 for 100 epochs. In the sentiment analysis stream, the batch size is 32, while in the churn prediction stream it is set to 16. The complete TCN model structure is shown in Fig. 3.

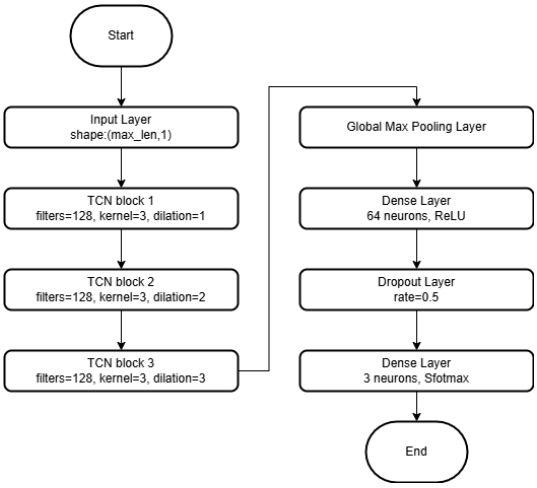


Fig 3. The structure of TCN model implementation.

4. Result and Discussion

This chapter presents the model performance evaluation results based on the previously described methodology. The primary evaluation focus is on accuracy to assess the effectiveness of the LSTM and TCN models in two analysis streams: sentiment and churn prediction. Additionally, precision, recall, and F1-score metrics are reported to provide a qualitative view of the models' ability to classify positive and negative classes. A confusion matrix is also presented to show the distribution of correct and incorrect predictions for each class.

4.1. LSTM Sentiment Model Evaluation

LSTM model performance in sentiment analysis was evaluated using a binary confusion matrix. The confusion matrix results for the test data with two classes (Negative and Positive) are presented in Table 5.

Table 5. LSTM Sentiment Confusion Matrix Results

Actual \ Predicted	Negative	Positive	Total
Negative	17,000	2,684	19,684
Positive	2,983	18,852	21,835
Total	19,983	21,536	41,519

From this confusion matrix, it can be seen that the LSTM model correctly detected the majority of negative reviews with 17,000 samples, but 2,684 negative reviews were misclassified as positive. Similarly, the model correctly identified 18,852 positive reviews, although 2,983 positive reviews were misclassified as negative. This error distribution shows the model's tendency to leverage stronger features from the majority class but still requires refinement to limit errors on the class transition pulse.

Table 6. LSTM Sentiment Classification Evaluation Matrix

Class	Precision	Recall	F1-score	Support
Negative	85.10%	86.35%	85.74%	19,684
Positive	87.50%	86.35%	86.90%	21,835
Macro average	86.30%	86.35%	86.32%	41,519
Weighted average	86.20%	86.35%	86.36%	41,519

Based on Table 6, overall get an accuracy of 86.35%%. The 85.10 % precision for the negative class indicates a relatively high proportion of correctly identified negative reviews, while the 86.35 % recall shows the model's ability to capture most negative instances. The 85.74 % F1-score reflects a balance between precision and recall for the negative class. For the positive class, the 87.50 % precision and 86.35 % recall yield an F1-score of 86.90 %, confirming balanced performance in avoiding both false positives and false negatives. The macro and weighted averages, which closely match the overall accuracy, confirm consistent model performance across both classes.

4.2. TCN Sentiment Model Evaluation

TCN model performance in sentiment analysis was evaluated using a binary confusion matrix. The confusion matrix results for the test data with two classes (Negative and Positive) are shown in Table 7.

Table 7. TCN Sentiment Confusion Matrix Results

Actual \ Predicted	Negative	Positive	Total
Negative	17,999	1,685	19,684
Positive	1,828	20,007	21,835
Total	19,827	21,692	41,519

From this confusion matrix, the TCN model correctly classified 17,999 negative reviews, although 1,685 negative reviews were misclassified as positive. For the positive class, 20,007 reviews were accurately identified, while 1,828 positive reviews were misclassified as negative. This error pattern indicates a challenge for the model in distinguishing comments that contain borderline phrases or contexts between negative and positive sentiment. Nevertheless, the majority of reviews are classified correctly, confirming the TCN model's ability to handle sequential text data.

Table 8. TCN Sentiment Classification Evaluation Matrix

Class	Precision	Recall	F1-score	Support
Negative	90.79%	91.41%	91.10%	19,684
Positive	92.23%	91.63%	91.93%	21,835
Macro average	91.51%	91.52%	91.52%	41,519
Weighted average	91.48%	91.55%	91.50%	41,519

Further analysis of Table 8 shows that the model achieved an overall accuracy of 91.55%. A 90.79% precision for the negative class reflects the ratio of correctly detected negative reviews to total negative predictions, while a 91.41% recall indicates the proportion of negative reviews successfully recognized by the model. The 91.10% F1-score represents the harmonic balance between precision and recall for this class. For the positive class, a 92.23% precision and 91.63% recall yield a 91.93% F1-score, indicating the model's ability to avoid high prediction errors on positive reviews. The macro average and weighted average values, which closely match the overall accuracy, confirm stable performance across both classes. Overall, these results highlight the TCN model's strength in managing data with low to moderate sequential text complexity without the need for additional techniques.

4.3. LSTM Transaction Model Evaluation

Model performance evaluation of the LSTM model in the churn-prediction stream was conducted using a binary confusion matrix. The confusion matrix results for the test data are presented in Table 9.

Table 9. LSTM Transaction Confusion Matrix Results

Actual \ Predicted	Non-Churn (0)	Churn (1)	Total
Non-Churn (0)	28	0	28
Churn (1)	6	11	17
Total	34	11	45

From 45 transaction samples, the LSTM model classified 39 samples correctly, achieving an accuracy of 86.67 %. Table 10 shows the classification evaluation metrics for both classes. In detail, Table 9 shows that all 28 non-churn customers were correctly identified, with no false positives. Conversely, 6 churn customers were not detected (false negatives), although 11 churn customers were correctly recognized. This pattern indicates that the model reliably detects non-churn status but requires improved sensitivity to churn events.

Table 10. LSTM Transaction Classification Evaluation Metrics

Class	Precision	Recall	F1-score	Support
Non-Churn	82.00%	100.00%	90.00%	28
Churn	100.00%	65.00%	79.00%	17
Macro average	91.00%	82.50%	84.50%	45
Weighted average	88.80%	86.89%	85.74%	45

In Table 10, shows that the model achieved an overall accuracy of 86.67%. The 82.00 % precision for the non-churn class indicates that of all non-churn predictions, 82.00 % were indeed non-churn customers, while the recall of 100.00 % shows that no non-churn customer was missed. For the churn class, the 100.00 % precision means every churn prediction corresponded to an actual churn customer, but the 65.00 % recall indicates 35.00 % of churn customers were not detected. The F1-scores are 90.00 % for non-churn and 79.00 % for churn.

The macro average of 84.50 % and the weighted average of 85.74 % reflect the model's balanced performance and consistency across both classes.

4.4. TCN Transaction Model Evaluation

TCN model performance in the churn-prediction stream was also evaluated using a binary confusion matrix. The confusion matrix results for the test data are shown in Table 11.

Table 11. TCN Transaction Confusion Matrix Results

Actual \ Predicted	Non-Churn (0)	Churn (1)	Total
Non-Churn (0)	26	2	28
Churn (1)	2	18	20
Total	28	20	48

From a total of 48 samples, the TCN model correctly classified 44 samples, achieving an accuracy of 91.67 %. Table 11 shows that of 28 non-churn customers, 26 were correctly identified (true negative) while 2 non-churn customers were misclassified as churn (false positive). In the churn class, 18 customers were correctly recognized (true positive) and 2 churn customers were not detected (false negative). This error pattern indicates that the TCN model is more reliable at identifying active customers but still needs improved sensitivity to potential churn.

Table 12. TCN Transaction Classification Evaluation Metrics

Class	**Precision**	**Recall**	**F1-score**	**Support**
Non-Churn	92.86%	92.86%	92.86%	28
Churn	90.00%	90.00%	90.00%	20
Macro average	91.43%	91.43%	91.43%	48
Weighted average	91.79%	91.67%	91.79%	48

Analysis of Table 12 shows that model achieved an overall accuracy of 91.67%. precision and recall for the non-churn class are both 92.86 %, indicating that most non-churn predictions are correct and most non-churn customers are identified. For the churn class, precision and recall of 90.00 % show that the model detects most churn customers with good accuracy despite 10 % false negatives. Both classes have F1-scores above 90 %, confirming a balance between precision and recall. Macro average and weighted average above 91 % underscore consistent TCN model performance across both classes. From a business perspective, this approach is effective for monitoring churn risk, but it is recommended to integrate further prevention mechanisms for customers with high churn probability.

4.5. Comparison of LSTM and TCN Performance

The performance comparison of the two models is presented in two tables that separate the results for the sentiment analysis task and the churn prediction task, making it easier to understand each model's strengths in different contexts.

Table 13. Summary of Metric Comparison for LSTM and TCM Models for Sentiment

Model	Accuracy	Precision Avg	Recall Avg	F1-score Avg
LSTM	86.35%	86.30%	86.35%	86.32%
TCN	91.55%	91.51%	91.52%	91.52%

In Table 13, the 5.20% point increase in TCN accuracy indicates the convolutional model's ability to capture long-term context and phrase patterns in review text. The differences in average precision and recall suggest that TCN is not only more precise but also more reliable in identifying all positive and negative instances. The balanced average F1-score confirms that these two metrics are aligned in TCN performance.

Table 14. Summary of Metric Comparison for LSTM And TCN Models for Churn Prediction

Model	Accuracy	Precision Avg	Recall Avg	F1-score Avg
LSTM	86.67%	91.00%	82.50%	86.32%
TCN	91.67%	91.43%	91.43%	91.43%

Table 14 reinforces TCN's dominance in churn prediction, especially in the recall metric, which reflects the model's ability to identify customers who are likely to stop transacting. The increase in average recall from 82.50 % (LSTM) to 91.43 % (TCN) means that TCN substantially reduces the number of false negatives, a critical factor for retention strategies. The slightly higher average precision of TCN indicates consistent accuracy in churn prediction.

Overall, this comparison demonstrates that TCN outperforms LSTM in both classification tasks because its convolutional architecture more effectively extracts temporal and spatial patterns. For practical deployment on the Baliyoni e-commerce platform, TCN is recommended as the primary model, while LSTM may remain useful in environments with computational constraints or when integration with long-memory based NLP pipelines is required.

4.6. Churn Probability Calculation

This churn probability analysis is based on the actual distribution of sentiment and churn-status combinations in the dataset. Table XV presents a summary of counts and percentages for each combination of sentiment (Negative or Positive) and churn status (Non-Churn or Churn).

Table 15. Summary of Sentiment and Churn Probabilities

Sentiment	Churn Status	Count	Percentage (%)
Negative	Non-Churn	1,768	47.30
Negative	Churn	20	0.54
Positive	Non-Churn	1,932	51.69
Positive	Churn	18	0.48

Next, the conditional churn probabilities are calculated for each sentiment category:

1. P(churn | negative sentiment) = 20 / (1,768 + 20) ≈ 1.12 %
2. P(non-churn | negative sentiment) = 1,768 / (1,768 + 20) ≈ 98.88 %
3. P(churn | positive sentiment) = 18 / (1,932 + 18) ≈ 0.92 %
4. P(non-churn | positive sentiment) = 1,932 / (1,932 + 18) ≈ 99.08 %

Further interpretation reveals that although the positive-sentiment group has a slightly larger overall share than the negative group (51.69 % versus 47.30 %), its churn risk (0.92 %) is lower than that of the negative group (1.12 %). This finding confirms that negative sentiment does not always lead to churn and highlights the importance of addressing issues raised in negative reviews to reinforce customer loyalty. From a practical perspective, management can use these insights to prioritise service-improvement initiatives based on the most critical customer complaints.

4.7. Results Validation and Expert Analysis

The evaluation results of the LSTM and TCN models in both the sentiment analysis stream and the churn prediction stream were reviewed by Ms. Ni Putu Lilik Mariasih, S.E., to ensure data validity and classification accuracy. In sentiment analysis, Ms. Lilik assessed that the LSTM model can classify both positive and negative reviews effectively, especially long reviews with complex language patterns, but it is sometimes less responsive to context variations in short reviews. In contrast, the TCN model was praised for its consistency and reliability in recognizing positive and negative sentiment even when reviews contain direct and brief phrases.

In churn prediction, the validation showed that TCN has higher sensitivity in detecting customers at risk of stopping transactions, as indicated by its superior recall value. The LSTM model remains effective in identifying churn patterns in historical transaction data with long sequences, but TCN was deemed more suitable for transactions that are concise and direct. Based on this validation, it can be concluded that LSTM is appropriate for tasks requiring deep understanding of long reviews and comprehensive transaction histories, whereas TCN is better suited for scenarios that demand speed and consistency in processing short text and transaction signals. A hybrid approach may also be considered to combine the strengths of both models.

5. Conclusion

The results of this study show that the Temporal Convolutional Network (TCN) model consistently outperforms the Long Short-Term Memory (LSTM) model in both analysis contexts tested. In the sentiment analysis task, TCN increased accuracy and the average precision, recall, and F1-score by more than five percentage points compared to LSTM, confirming this convolutional architecture's ability to capture long-range structure and context in customer review text. In transaction churn prediction, TCN again achieved the highest accuracy of 91.67% and a higher recall, indicating that this model is more reliable at detecting customers who are likely to stop transacting and thus can support more effective retention strategies.

Conditional probability analysis shows that customers with negative sentiment who remain active dominate the data, representing almost 50% of the sample and having a non-churn probability of nearly 99%. By contrast, the combination of positive sentiment and churn occurred in less than 1% of samples, indicating that negative expression does not necessarily lead to customer loss. These findings underscore the importance of prioritising service-improvement initiatives based on negative feedback, since these customers still demonstrate significant loyalty potential.

Given its superior performance and stability, TCN is recommended as the primary model for churn detection and sentiment analysis on the Baliyoni Group e-commerce platform. LSTM remains useful in scenarios with limited computational resources or when long-term memory features are a priority. Overall, the integration of sentiment analysis, the RFM method, and deep-learning models provides a comprehensive framework for understanding customer behaviour and designing targeted retention interventions. Future work applying ensemble methods and further hyperparameter optimisation is expected to improve prediction accuracy and extend analysis to longer sequential data.

Acknowledgement

We gratefully acknowledge BINUS University for supplying the resources and support essential to this study. We also thank Baliyoni Group (Balimall and Tokodaring) for providing access to the customer review dataset that formed the basis of our analysis. We are

especially grateful to Mrs Ni Putu Lilik Mariasih, S E., whose expert review greatly bolstered the credibility of our findings. Finally, we appreciate the feedback from colleagues and peer reviewers, whose insights helped us refine this work.

References

[1] K. HAJI, "E-commerce development in rural and remote areas of BRICS countries," *J. Integr. Agric.*, vol. 20, no. 4, pp. 979–997, 2021, doi: 10.1016/S2095-3119(20)63451-7.
[2] M. Xi, Z. Luo, N. Wang, and J. Yin, "A Latent Feelings-aware RNN Model for User Churn Prediction with Behavioral Data," Nov. 2019.
[3] Y. Gaidhani *et al.*, "AI-Driven Predictive Analytics for CRM to Enhance Retention Personalization and Decision-Making," *Int. J. Adv. Comput. Sci. Appl.*, vol. 16, no. 4, pp. 552–563, 2025, doi: 10.14569/IJACSA.2025.0160456.
[4] C. G. Mena, A. De Caigny, K. Coussement, K. W. De Bock, and S. Lessmann, "Churn Prediction with Sequential Data and Deep Neural Networks. A Comparative Analysis," Sep. 2019.
[5] G. Mena, K. Coussement, K. W. De Bock, A. De Caigny, and S. Lessmann, "Exploiting time-varying RFM measures for customer churn prediction with deep neural networks," *Ann. Oper. Res.*, vol. 339, no. 1–2, pp. 765–787, 2024, doi: 10.1007/s10479-023-05259-9.
[6] W. Wu, Y. Zhang, and Y. Fan, "ICT Empowers the Formation and Development of Rural E-Commerce in China," *IEEE Access*, vol. 8, pp. 135264–135283, 2020, doi: 10.1109/ACCESS.2020.3011593.
[7] S. Zhang, D. Zhang, H. Zhong, and G. Wang, "A multiclassification model of sentiment for e-commerce reviews," *IEEE Access*, vol. 8, pp. 189513–189526, 2020, doi: 10.1109/ACCESS.2020.3031588.
[8] G. Li, Q. S. Zheng, L. Zhang, S. Z. Guo, and L. Y. Niu, "Sentiment Infomation based Model for Chinese text Sentiment Analysis," in *2020 IEEE 3rd International Conference on Automation, Electronics and Electrical Engineering, AUTEEE 2020*, Nov. 2020, pp. 366–371. doi: 10.1109/AUTEEE50969.2020.9315668.
[9] D. Putrisia and K. M. Lhaksmana, "El Niño Sentiment Analysis Using Recurrent Neural Network and Convolutional Neural Network Use GloVe," *Technol. Sci.*, vol. 6, no. 1, 2024, doi: 10.47065/bits.v6i1.5284.
[10] Y. Azzery, "Analysis of E-commerce Growth in the Industrial Age 4.0 in Indonesia," *Int. J. Eng. Contin.*, vol. 1, 2022, doi: 10.58291/ijec.v1n1.33.
[11] H. Kim and G. Qin, "Summarizing Students' Free Responses for an Introductory Algebra-Based Physics Course Survey Using Cluster and Sentiment Analysis," *IEEE Access*, vol. 11, pp. 89052–89066, 2023, doi: 10.1109/ACCESS.2023.3305260.
[12] R. Li, Z. Ouyang, Z. Shang, L. Jia, and X. Li, "Learning Common and Label-Specific Features for Multi-Label Classification With Missing Labels," *IEEE Access*, vol. 12, pp. 81170–81195, 2024, doi: 10.1109/ACCESS.2024.3411095.
[13] H. Kour and M. K. Gupta, "Hybrid LSTM-TCN Model for Predicting Depression using Twitter Data," *2022 17th Int. Conf. Control. Autom. Robot. Vis.*, pp. 167–172, 2022, doi: 10.1109/ICARCV57592.2022.10004238.
[14] M. Deng, "Machine Learning Advances in Technology Applications: Cultural Heritage Tourism Trends in Experience Design," *Int. J. Adv. Comput. Sci. Appl.*, vol. 16, no. 4, pp. 186–196, 2025, doi: 10.14569/IJACSA.2025.0160420.
[15] D. Cao, Y. Huang, H. Li, X. Zhao, Q. Zhao, and Y. Fu, "Text Sentiment Classification Based on LSTM-TCN Hybrid Model and Attention Mechanism," *Proc. 4th Int. Conf. Comput. Sci. Appl. Eng.*, 2020, doi: 10.1145/3424978.3425092.
[16] H. Ghallab, M. Nasr, and H. Fahmy, "Mitigating Catastrophic Forgetting in Continual Learning Using the Gradient-Based Approach: A Literature Review," *Int. J. Adv. Comput. Sci. Appl.*, vol. 16, no. 4, pp. 120–130, 2025, doi: 10.14569/IJACSA.2025.0160414.
[17] B. K. Shrivash, D. K. Verma, and P. Pandey, "An Effective Framework for Sentiment Analysis Using RNN and LSTM-Based Deep Learning Approaches," in *ICACDS*, 2023. doi: 10.1007/978-3-031-37940-6_28.
[18] X. Wen and W. Li, "Time Series Prediction Based on LSTM-Attention-LSTM Model," *IEEE Access*, vol. 11, pp. 48322–48331, 2023, doi: 10.1109/ACCESS.2023.3276628.
[19] F. Wu, J. Liu, J. Yang, L. Zhang, Y. He, and Z. Lin, "Learning multiband-Temporal-spatial EEG representations of emotions using lightweight temporal convolution and 3D convolutional neural network," *IEEE Access*, 2024, doi: 10.1109/ACCESS.2024.3460393.
[20] L. Hickman, S. Thapa, L. Tay, M. Cao, and P. Srinivasan, "Text Preprocessing for Text Mining in Organizational Research: Review and Recommendations," *Organ. Res. Methods*, vol. 25, pp. 114–146, 2020, doi: 10.1177/1094428120971683.
[21] F. Yulidayanti, Safwandi, "Sentiment Analysis of Free Online Novel Applications Using the Support Vector Machine Method," *Int. J. Eng. Sci. Inf. Technol.*, vol. 5, no. 1, pp. 340–349, 2025, doi: 10.52088/ijesty.v5i1.732.
[22] Y. A. Tursina Dewi, Asrianda Asrianda, "Sentiment Analysis of Customer Satisfaction Towards Shopee and Lazada E-commerce Platform Using the Random Forest Algorithm Classifier," *Int. J. Eng. Sci. Inf. Technol.*, vol. 5, no. 1, pp. 229–235, 2025, doi: 10.52088/ijesty.v5i1.692.
[23] D. A. Naik, S. Mythreyan, and S. Seema, "Relevance Feature Discovery in Text Mining Using NLP," *2022 3rd Int. Conf. Emerg. Technol.*, pp. 1–6, 2022, doi: 10.1109/incet54531.2022.9824807.
[24] C. P. Chai, "Comparison of text preprocessing methods," *Nat. Lang. Eng.*, vol. 29, pp. 509–553, 2022, doi: 10.1017/S1351324922000213.
[25] F. Alshanik, A. W. Apon, A. Herzog, I. Safro, and J. Sybrandt, "Accelerating Text Mining Using Domain-Specific Stop Word Lists," *2020 IEEE Int. Conf. Big Data (Big Data)*, pp. 2639–2648, 2020, doi: 10.1109/BigData50022.2020.9378226.
[26] P. Bratke, R. Zimmermann, and R. Blum, "Fostering text mining with knowledge graphs: An approach to support business experts in defining domain-specific document sets," *Hum. Side Serv. Eng.*, 2023, doi: 10.54941/ahfe1003128.
[27] A. Peri\vsić and M. Pahor, "RFM-LIR Feature Framework for Churn Prediction in the Mobile Games Market," *IEEE Trans. Games*, vol. 14, pp. 126–137, 2022, doi: 10.1109/TG.2021.3067114.
[28] Y. T. Naing, M. Raheem, and N. K. Batcha, "Feature Selection for Customer Churn Prediction: A Review on the Methods \& Techniques applied in the Telecom Industry," *2022 IEEE Int. Conf. Distrib. Comput. Electr. Circuits Electron.*, pp. 1–5, 2022, doi: 10.1109/icdcece53908.2022.9793315.
[29] H. Hendro, A. M. Shiddiqi, and A. Saikhu, "Feature Weighting using Gravitational Search Algorithm in Customer Churn Prediction," *Proc. Int. Conf. Comput. Mach. Learn. Data Sci.*, 2024, doi: 10.1145/3661725.3661787.

[30] M. Hao, "Research on Customer Churn Prediction Based on PSO-SA Feature Selection Algorithm," *2024 5th Int. Semin. Artif. Intell. Netw. Inf. Technol.*, pp. 1055–1059, 2024, doi: 10.1109/AINIT61980.2024.10581724.
[31] J. Dias, "Machine Learning in Bank Customer Churn Prediction: Improving Accuracy through Feature Engineering," *Int. J. Res. Appl. Sci. Eng. Technol.*, 2025, doi: 10.22214/ijraset.2025.67002.
[32] N. Alboukaey, A. Joukhadar, and N. Ghneim, "Dynamic behavior based churn prediction in mobile telecom," *Expert Syst. Appl.*, vol. 162, p. 113779, 2020, doi: 10.1016/j.eswa.2020.113779.
[33] Q. Yang, "Research on E-commerce Customer Satisfaction Evaluation Method Based on PSO-LSTM and Text Mining," *3C Empres. Investig. y Pensam. crítico*, vol. 12, no. 01, pp. 51–66, Mar. 2023, doi: 10.17993/3cemp.2023.120151.51-66.
[34] S. Abdul-Rahman, M. F. A. M. Ali, A. A. Bakar, and S. Mutalib, "Enhancing churn forecasting with sentiment analysis of steam reviews," *Soc. Netw. Anal. Min.*, vol. 14, p. 178, 2024, doi: 10.1007/s13278-024-01337-3.
[35] G. Kazbekova, Z. Ismagulova, G. Ibrayeva, A. Sundetova, Y. Abdrazakh, and B. Baimurzayev, "Real-Time Lightweight Sign Language Recognition on Hybrid Deep CNN-BiLSTM Neural Network with Attention Mechanism," *Int. J. Adv. Comput. Sci. Appl.*, vol. 16, no. 4, pp. 510–522, 2025, doi: 10.14569/IJACSA.2025.0160452.
[36] D. H. Rudd, H. Huo, M. R. Islam, and G. Xu, "Churn Prediction via Multimodal Fusion Learning: Integrating Customer Financial Literacy, Voice, and Behavioral Data," *2023 10th Int. Conf. Behav. Soc. Comput.*, pp. 1–7, 2023, doi: 10.1109/BESC59560.2023.10386253.
[37] E. A. el Kassem, S. A. Hussein, A. H. M. Abdelrahman, and F. K. Alsheref, "Customer Churn Prediction Model and Identifying Features to Increase Customer Retention based on User Generated Content," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, 2020, doi: 10.14569/ijacsa.2020.0110567.