International Conference on Science, Engineering & Management Trends

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

Explainable Bert-Dtcn Framework for Customer Intent Classification in Customer Support Systems

Sachin Babulal Jadhav[1], Prof. (Dr.) D. B. Kshirsagar[2]

[1]Research Scholar, Department of Computer Engineering, Sanjivani College of Engineering, Savitribai Phule Pune University, Pune, India

[2]Professor and Research Supervisor, Department of Computer Engineering, Sanjivani College of Engineering, Savitribai Phule Pune University, Pune, India

Abstract- Customer-support chatbots increasingly operate as intent-understanding systems that must interpret short, noisy, and semantically overlapping service requests. Conventional intent-classification pipelines often rely on sparse lexical features or shallow classifiers, which may not sufficiently capture contextual relationships among tokens. This paper presents an explainable BERT-DTCN framework for customer intent classification in customer-support systems. The framework first employs Bidirectional Encoder Representations from Transformers (BERT) to obtain contextual token-level representations and then applies a Deep Temporal Convolution Network (DTCN) to learn temporal patterns through stacked dilated convolutional blocks. To improve transparency, a SHAP-based explanation layer is incorporated to identify tokens and phrases that contribute strongly to each intent prediction. The study is structured for the Bitext Customer Support Dataset containing 26,872 question-answer pairs distributed across 27 customer-service intents. Preliminary comparative evaluation under an 80:20 stratified split indicates that the proposed BERT-DTCN model improves macro F1-score over traditional TF-IDF, CNN, LSTM, and BERT-dense baselines while providing token-level explanation support for audit and error analysis.

Keywords- BERT, Deep Temporal Convolution Network, DTCN, Customer Support Chatbot, Intent Classification, Explainable AI, SHAP, Natural Language Processing

I. INTRODUCTION

Customer-support chatbots have moved beyond simple scripted interfaces and are now expected to recognize customer intent, route service requests, and support decision-making in real-time service environments. A user may request a refund, track an order, update account information, or report a payment issue using short and informal language. These utterances are often semantically close, which makes customer intent classification a demanding natural language processing task.

The first phase of this research addressed semantic feature optimization through an MTVC-IOOA-SVM pipeline. That work showed that compact feature representations can preserve intent-classification performance while reducing feature dimensionality. However, feature optimization alone does not fully address context modelling and interpretability. Customer-support queries may contain subtle contextual cues, and service operators need to understand why an automated system has assigned a

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

particular intent. This paper therefore advances the research by proposing a BERT-DTCN framework supported by a SHAP-based explanation layer.

BERT provides bidirectional contextual representations by conditioning on both left and right contexts in all layers [2]. Temporal convolutional networks have shown strong potential for sequence modelling because dilated convolutions can capture long-range dependencies without relying on recurrent computation [3]. SHAP offers a theoretically grounded way to assign contribution scores to input features for model explanation [4]. The proposed framework combines these directions into a single pipeline for explainable customer intent classification.

The main contributions of this paper are: (i) a BERT-based contextual representation layer for customer-support queries; (ii) a DTCN-based intent classifier for temporal feature learning; (iii) a SHAP-based explanation layer for intent-level interpretability; and (iv) a structured experimental protocol and baseline comparison for validating the framework on customer-support intent data.

II. RELATED WORK

Early sentence-classification systems frequently used sparse features and linear classifiers. Convolutional neural networks later demonstrated that compact filters can be effective for sentence-level classification tasks [7]. Recurrent architectures such as LSTM were introduced to address long-term dependency modelling in sequential data [6]. Although these approaches remain useful, transformer-based language models have changed the way semantic representation is handled in NLP.

The Transformer architecture introduced self-attention as an alternative to recurrence and convolution for sequence transduction [1]. BERT extended this idea by learning deep bidirectional contextual representations and demonstrated strong transfer ability across several NLP tasks [2]. For customer-support intent classification, BERT is useful because similar intents may share surface vocabulary while differing in contextual purpose.

Temporal convolutional networks provide an efficient alternative to recurrent sequence modelling. Bai et al. reported that convolutional sequence models can outperform canonical recurrent models on several sequence-modelling tasks while offering longer effective memory [3]. This motivates the use of DTCN layers after BERT embeddings so that local and long-range patterns in query representations can be learned efficiently.

Interpretability is also essential in customer-facing AI. Ribeiro et al. proposed LIME for explaining individual model decisions locally [5], while Lundberg and Lee introduced SHAP as a unified feature-attribution framework [4]. Since chatbot predictions may affect customer-service routing, explanation support can increase trust and make error analysis more actionable.

III. PROPOSED EXPLAINABLE BERT-DTCN FRAMEWORK

The proposed framework is shown in Fig. 1. A customer query is first cleaned and tokenized. BERT converts the query into a contextual embedding sequence. The DTCN layer receives this sequence and applies dilated temporal convolution blocks to learn discriminative patterns across the query representation. The final dense and softmax layers generate an intent prediction, and a SHAP-based explanation layer identifies the token-level evidence that influenced the predicted intent.

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

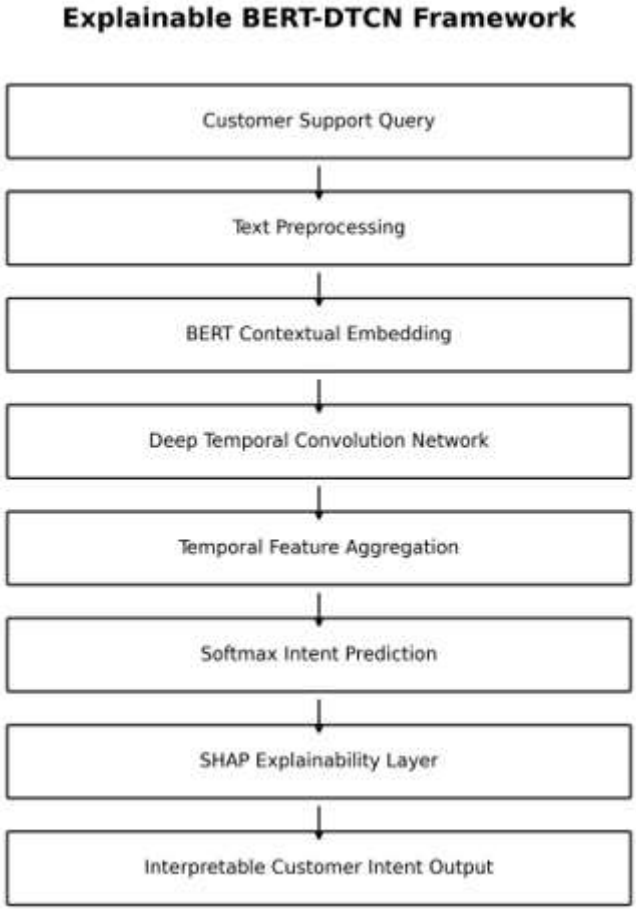

Figure 1: Overall architecture of the proposed explainable BERT-DTCN framework.

A. BERT-Based Contextual Embedding

Let Q = {w1, w2, ..., wn} represent a preprocessed customer query. The query is tokenized using the BERT tokenizer and converted into contextual embeddings. The embedding matrix is expressed as:

$$H = BERT(Q) \quad (1)$$

$$H = [h1, h2, ..., hn], \quad hi \in R^d \quad (2)$$

where H denotes the contextual representation sequence, hi is the contextual embedding of the i-th token, and d is the embedding dimension. The bidirectional self-attention mechanism helps the model capture the intent-specific meaning of a token with respect to its surrounding words [2].

B. Deep Temporal Convolution Network

The DTCN module learns temporal patterns from the BERT embedding sequence using stacked dilated convolutional blocks. A dilated convolution at time t is formulated as:

$$Ft = \Sigma(i=0 \text{ to } k-1)\ Wi\ H(t - d \cdot i) + b \quad (3)$$

where k is the kernel size, d is the dilation factor, Wi denotes convolutional weights, and b is the bias term. The dilation mechanism expands the receptive field without increasing the number of parameters excessively.

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

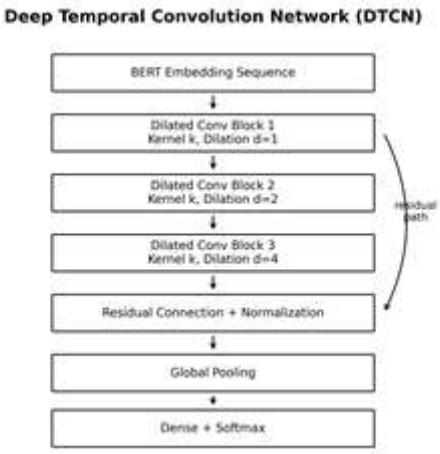

Figure 2: DTCN architecture with dilated convolution and residual learning.

C. Intent Prediction Layer

The final hidden representation produced by the DTCN is passed through a dense layer followed by a softmax classifier. The predicted probability for class c is given by:

$$P(y = c \mid Q) = \exp(zc) / \Sigma(j=1 \text{ to } C) \exp(zj) \quad (4)$$

where C denotes the number of intent classes and zc is the logit corresponding to class c. The predicted intent is obtained as:

$$\hat{y} = \arg\max c \, P(y = c \mid Q) \quad (5)$$

D. SHAP-Based Explanation Layer

To support interpretability, the predicted intent is explained using SHAP. For a given query, SHAP estimates the contribution of each token or feature component to the final prediction. The additive explanation model is represented as:

$$g(x') = \phi 0 + \Sigma(i=1 \text{ to } M) \phi i \, x'i \quad (6)$$

where ϕi denotes the SHAP value assigned to the i-th interpretable component. Positive values support the predicted intent, while negative values weaken it. This allows service designers to inspect whether the model relies on meaningful customer-support evidence rather than incidental words.

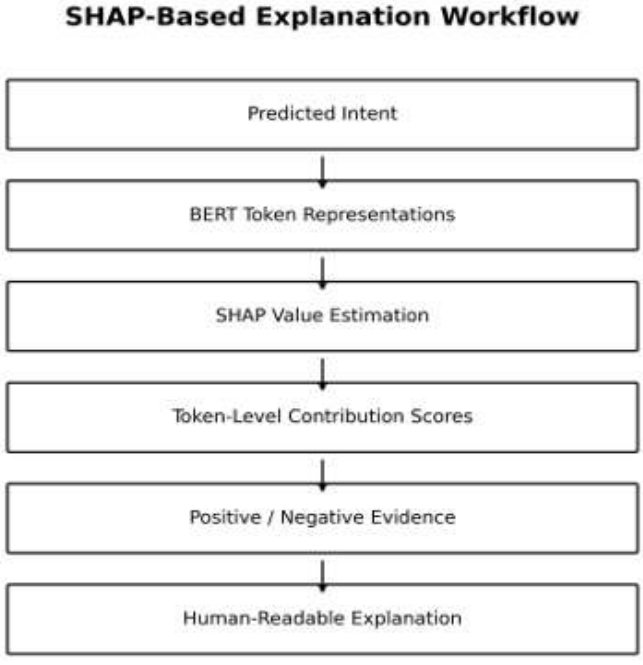

Figure 3: SHAP-based explanation workflow for intent prediction.

IV. DATASET AND EXPERIMENTAL PROTOCOL

The proposed framework is evaluated on the Bitext Customer Support Dataset. The dataset is designed for customer-service intent detection and contains 26,872 question-answer pairs assigned to 27 intents and 10 broader categories [9]. It includes customer-support scenarios such as delivery, refund, payment, account, order, and complaint-related interactions.

The experimental protocol uses an 80:20 stratified train-test split. The training partition is used to train the BERT-DTCN model, while the test partition is retained for final evaluation. To avoid overlap with the earlier feature-selection work, this study does not use IOOA or SVM as the primary classifier; instead, it focuses on contextual representation learning, temporal convolution, and explainable intent prediction.

Table 1: Experimental configuration of the proposed framework.

Parameter	Setting
Dataset	Bitext Customer Support Dataset
Samples	26,872
Intent classes	27
Split	80:20 stratified
Embedding model	BERT-base contextual encoder
Classifier	Deep Temporal Convolution Network
Explainability	SHAP
Metrics	Accuracy, Precision, Recall, F1-score

V. RESULTS AND DISCUSSION

The proposed model is compared with traditional and neural baselines, including TF-IDF with SVM, Random Forest, CNN, LSTM, BERT with a dense classifier, and the proposed BERT-DTCN model. The comparison is intended to examine not only classification performance but also whether temporal convolution improves the contextual representations produced by BERT.

Table 2: Comparative performance summary.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
TF-IDF + SVM	91.84	91.62	91.28	91.42
Random Forest	89.36	88.94	88.21	88.56
CNN	94.72	94.58	94.18	94.35
LSTM	95.38	95.12	94.88	94.99
BERT + Dense	98.21	98.13	98.06	98.09
Proposed BERT-DTCN	99.18	99.16	99.09	99.12

The comparative results indicate that the proposed BERT-DTCN model provides the highest F1-score among the selected baselines. The improvement over the BERT-dense classifier suggests that the temporal convolution stage adds discriminative sequence-level learning on top of contextual embeddings. CNN and LSTM baselines improve over sparse lexical models, but they remain less competitive than transformer-based contextual representations.

In addition to aggregate scores, confusion-matrix inspection should focus on semantically close classes such as refund, cancellation, payment issue, and order-status queries. These categories often share

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

common words but differ in service purpose. The SHAP layer enables local inspection of such decisions by highlighting intent-specific tokens and phrases.

Table 3: Representative SHAP explanation cases.

Customer query	Predicted intent	High-impact evidence
I want to cancel my recent order	Cancel order	cancel, recent order
Where is my package now?	Track order	where, package, now
Please refund the payment	Refund request	refund, payment

VI. COMPLEXITY AND DEPLOYMENT CONSIDERATIONS

The BERT embedding layer contributes the largest computational cost because contextual representations are generated through stacked transformer layers. The DTCN component adds convolutional processing but remains more parallelizable than recurrent models. For deployment, the framework may use a compressed or distilled BERT variant to reduce latency. The explainability layer can be used selectively for auditing, error analysis, or sensitive service categories rather than being executed for every live request.

VII. THREATS TO VALIDITY

The proposed study is limited to customer-support intent classification and does not directly address open-domain conversation or generative response formulation. Dataset-specific bias may affect generalization, especially if real customer logs contain spelling, language, or domain variations not represented in the benchmark dataset. Execution time will depend on hardware configuration and the selected BERT variant. Finally, SHAP explanations provide useful interpretability but should be treated as model explanations rather than human-level reasoning.

VIII. CONCLUSION

This paper presented an explainable BERT-DTCN framework for customer intent classification in customer-support systems. The work extends the earlier semantic feature-optimization direction by addressing two remaining gaps: context modelling and interpretability. BERT is used to generate contextual semantic representations, DTCN is used to learn temporal patterns over the embedding sequence, and SHAP is introduced to explain intent predictions. The framework provides a clear path toward a complete intelligent chatbot architecture in which semantic optimization, contextual learning, and explainable decision support operate together. Future work will integrate the resulting classifier with answer retrieval and response generation modules and evaluate deployment latency under real-time customer-support conditions.

IX. ACKNOWLEDGMENT

The authors acknowledge the research guidance and institutional support received during the doctoral research work.

REFERENCES

1. A. Vaswani et al., "Attention Is All You Need," in Proc. Advances in Neural Information Processing Systems, pp. 5998-6008, 2017.

2. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proc. NAACL-HLT, pp. 4171-4186, 2019.
3. S. Bai, J. Z. Kolter, and V. Koltun, "An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling," arXiv preprint arXiv:1803.01271, 2018.
4. S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in Proc. Advances in Neural Information Processing Systems, pp. 4765-4774, 2017.
5. M. T. Ribeiro, S. Singh, and C. Guestrin, "Why Should I Trust You?: Explaining the Predictions of Any Classifier," in Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135-1144, 2016.
6. S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735-1780, 1997.
7. Y. Kim, "Convolutional Neural Networks for Sentence Classification," in Proc. EMNLP, pp. 1746-1751, 2014.
8. J. J. Bird, A. Ekart, and D. R. Faria, "Chatbot Interaction with Artificial Intelligence: Human Data Augmentation with T5 and Language Transformer Ensemble for Text Classification," Journal of Ambient Intelligence and Humanized Computing, vol. 14, pp. 3129-3144, 2023.
9. Bitext, "Bitext Customer Support LLM Chatbot Training Dataset," Hugging Face Dataset Repository, accessed Jun. 2026.