AI-Powered Ticket Insights: Bridging NLP and Ensemble Learning for Smarter Support

I. V. Prakash[1], Ale Abhi[1], Gudtlalwar Chetan[1], Parthade Kethan[1], A. Salman[1], Mynam Vijay Raja[1]
[1]Department of Electronics and Communication Engineering, [1]Sree Dattha Institute of Engineering and Science, Sheriguda, Ibrahimpatnam, 501510, Telangana, India.

To Cite this Article
I. V. Prakash, Ale Abhi, Gudtlalwar Chetan, Parthade Kethan, A. Salman, Mynam Vijay Raja, "AI-Powered Ticket Insights: Bridging NLP and Ensemble Learning for Smarter Support", Journal of Science Engineering Technology and Management Science, Vol. 03, Issue 06, June 2026, pp: 696-707, DOI: http://doi.org/10.64771/jsetms.2026.v03.i06.pp696-707
Submitted: 08-05-2026 *Accepted: 15-06-2026* *Published: 22-06-2026*

ABSTRACT
Customer support centers produce a significant amount of textual data through support tickets, which contain valuable information about customer problems, service urgency, and satisfaction levels. Extracting insights from this unstructured data is crucial for enhancing service performance and operational effectiveness. However, conventional methods often depend on manual analysis or basic machine learning approaches such as bag-of-words and Term Frequency–Inverse Document Frequency (TF-IDF). These techniques lack the ability to understand context and semantic relationships, leading to lower accuracy, limited scalability, and difficulty in handling multiple prediction tasks at once, ultimately impacting the quality and speed of customer service. To overcome these challenges, this study proposes a comprehensive analytical framework that combines natural language processing (NLP) with advanced machine learning techniques. The textual data is first cleaned and standardized using common NLP preprocessing steps, including tokenization, removal of stopwords, and lemmatization. Next, transformer-based feature extraction is performed using the eXtreme Language Network (XLNet), which generates contextual embeddings that capture deeper semantic meaning within the text. These embeddings provide a more informative numerical representation, improving the effectiveness of downstream analysis. Various machine learning models, such as Quadratic Discriminant Analysis (QDA), Linear Discriminant Analysis (LDA), Histogram-Based Gradient Boosting (HGB), Stochastic Gradient Descent (SGD), and Nearest Centroid (NC), are then trained on these features to perform multi-target classification. The framework predicts important outcomes, including ticket priority, customer satisfaction scores, and resolution status. In addition, exploratory data analysis is conducted to identify patterns and trends in the dataset. Overall, this integrated approach improves prediction performance, enables scalable processing, and supports more efficient, data-driven decision-making in customer support operations.

Keywords: Customer Support Analytics, Ticket Priority Prediction, Multi-Target Classification, Machine Learning, Natural Language Processing (NLP).

1. INTRODUCTION

The domain of ticket management and monitoring is rapidly evolving with the rise of Large Language Models (LLMs) and Generative AI technologies. These innovations make it possible to automate essential processes such as ticket categorization, priority assignment, and recommendation of solutions in a more intelligent and context-aware manner. By leveraging historical support records along with predefined Standard Operating Procedures (SOPs), these systems can significantly reduce the reliance on manual effort. This leads to faster response times and enhances overall efficiency within customer support operations.

Despite these advancements, implementing AI-powered ticket management systems introduces several real-world challenges. Organizations need to manage the transition of experienced personnel to new AI-enabled platforms while ensuring that critical domain knowledge and intellectual property are retained. Additionally, migrating key operational data such as Service Level Agreements (SLAs) and expert-defined rules—must be handled with precision to avoid disruptions. Addressing these concerns requires adaptable and scalable system designs. In response, this work proposes a modular microservices-based architecture that is independent of specific applications. The framework supports core functionalities including ticket clustering, prioritization, resolution support, data ingestion, and visualization. Such an approach allows organizations to incrementally and flexibly integrate AI capabilities into their existing workflows.

In real-world IT Service Management (ITSM) environments, effective incident handling extends beyond textual data analysis. It requires incorporating spatial information such as system topology and infrastructure relationships, as well as temporal patterns related to incident occurrence and progression. Additionally, managing data consistency and provenance across heterogeneous sources is crucial. Given the continuously evolving nature of IT environments, clustering approaches must also be adaptive, allowing dynamic adjustment of cluster structures as new data emerges. To address these complexities, this research proposes a context-aware framework that combines semantic similarity techniques with spatial and temporal insights for enhanced ticket clustering. By capturing the multifaceted characteristics of IT operations, the proposed approach improves the accuracy of incident grouping and prioritization. This leads to more effective diagnostics, better resource utilization, enhanced system stability, and improved customer satisfaction. Ultimately, the study advances the capabilities of ITSM ticket analysis by integrating LLM-based semantic intelligence with real-world operational context.

2. RELATED WORK

This section reviews significant research contributions in incident management, natural language processing (NLP), clustering techniques, and AI-driven resolution systems. The discussion is organized sequentially to highlight the evolution from traditional approaches to modern intelligent solutions in ticket management systems.

2.1 Traditional Incident Management Systems

Incident management plays a vital role in IT Service Management (ITSM), focusing on restoring normal operations quickly while minimizing business disruptions. Traditional systems, guided by the ITIL framework [5], rely heavily on structured workflows and manual ticket handling. Although effective in controlled environments, these approaches struggle to manage large volumes of tickets and complex issue categorization, resulting in delays and inefficiencies.

Accurate ticket classification and routing are critical for ensuring timely resolution, maintaining customer satisfaction, and meeting Service-Level Agreements (SLAs) [9]. However, improper routing often leads to inefficient resource allocation and increased operational costs [14]. To address these challenges, Ticket Automation (TA) systems have been introduced to streamline the process from ticket submission to resolution. A key component of these systems is automated ticket classification, which has become increasingly important due to the growing volume of support requests, particularly in large IT organizations [3].

2.2 NLP-Based Ticket Classification

With the advancement of NLP techniques, research has shifted toward improving automated ticket classification. Leimeister et al. [17] proposed a method that enhances classification performance by incorporating hierarchical label structures into the learning process. Their approach utilizes a pre-trained BERT model, demonstrating improved accuracy in ticket categorization.

Their findings highlight that the choice of embedding strategy significantly impacts classification results, emphasizing the importance of contextual representations in NLP tasks. In practical systems, incident classification typically involves assigning both priority (urgency) and type (nature of the issue), such as distinguishing between service requests and incident reports [12].

2.3 Clustering Techniques for Incident Management

Clustering algorithms play a crucial role in grouping similar tickets, enabling faster resolution and improved resource allocation. Jain et al. [11] provided a comprehensive review of clustering methods, emphasizing the importance of similarity measures in achieving effective grouping.

Huang et al. [10] introduced an adaptive clustering approach tailored for ITSM environments, incorporating temporal patterns and incident similarities. Their method demonstrated that dynamic clustering significantly improves efficiency in handling recurring incidents.

Hierarchical clustering is particularly useful in scenarios involving networked systems, where understanding relationships between incidents is essential. Murtagh and Contreras [13] explored hierarchical clustering methods and their advantages in capturing structured relationships within data.

2.4 Transformer Models and Embedding Techniques

The development of transformer-based models has significantly advanced NLP applications. Devlin et al. [7] introduced BERT, which revolutionized text representation by capturing contextual relationships within data. This advancement has led to the adoption of embedding techniques, such as those used in modern models like BGE, which effectively represent semantic and hierarchical relationships in textual data.

These embeddings enable more accurate classification and clustering of incident tickets, forming the foundation for intelligent incident management systems.

2.5 AI-Driven Incident Resolution Systems

Recent research has focused on integrating AI models to automate not only classification but also resolution generation. Brown et al. [4] introduced GPT-3, demonstrating its capability in handling complex NLP tasks, including text generation.

Subsequent work by Radford et al. [15] explored the application of GPT-based models in incident resolution, showing that AI systems can generate context-aware and accurate solutions based on categorized ticket information. This marks a shift toward fully automated incident management systems.

2.6 Knowledge-Based Systems in Incident Management

Knowledge bases play a critical role in supporting automated incident resolution by storing structured information about past issues and solutions. Aamodt and Nygård [1] discussed the application of case-based reasoning in decision-support systems, highlighting its effectiveness in reusing past solutions for similar problems.

More recent studies, such as those by Fensel et al. [8], emphasize the importance of structured and well-maintained knowledge bases in improving the accuracy, consistency, and efficiency of AI-driven ITSM systems.

3. PROPOSED SYSTEM

The proposed approach introduces a well-organized and intelligent framework for analyzing customer support data by combining advanced natural language processing with machine learning techniques. It follows a step-by-step pipeline that starts with data collection, followed by preprocessing and extraction of meaningful textual features. This is then succeeded by classification using multiple models and predictive analysis. To achieve a deeper understanding of the text, transformer-based feature extraction is applied, enabling the system to capture contextual and semantic relationships more effectively than traditional techniques. Various machine learning algorithms are utilized for classification tasks, allowing performance comparison across models and supporting better decision-making. In addition, the methodology incorporates evaluation and visualization components to measure and interpret model performance, as illustrated in Fig 1. A graphical user interface is also included to provide smooth user interaction, real-time processing, and an intuitive user experience.

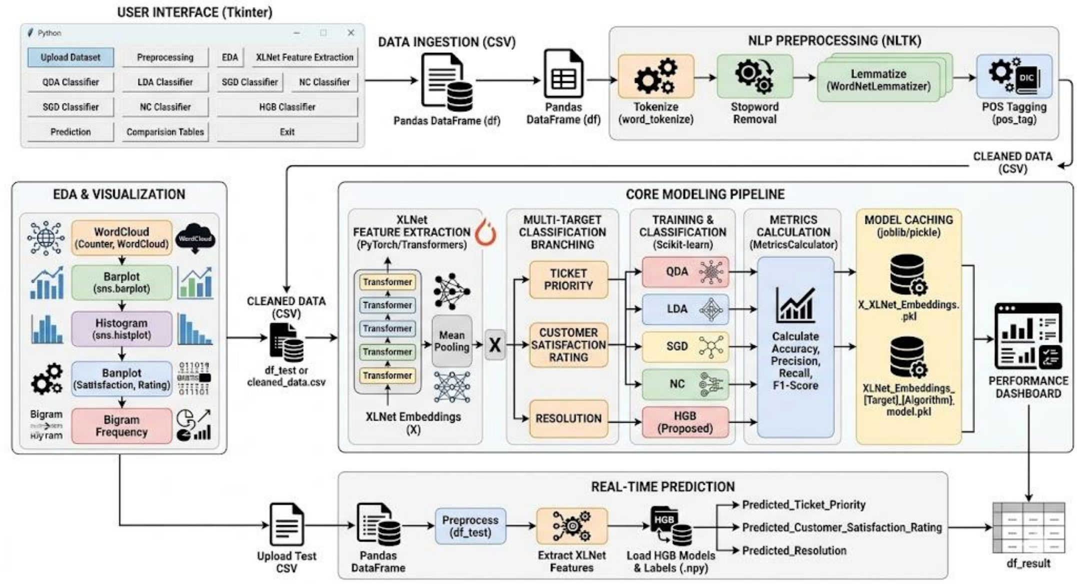

Fig. 1: Proposed system architecture.

User Interface (Tkinter GUI)

- The user interacts with the system through a dedicated graphical interface built using Tkinter, providing a centralized dashboard for support analytics.
- It provides a comprehensive suite of operational modules, including dataset upload, NLP preprocessing, EDA visualization, feature extraction, and model training.
- All user actions from triggering a prediction to comparing results are captured through intuitive button clicks and translated into backend function calls.
- The interface is designed to bridge the gap between complex machine learning workflows and the end-user's need for actionable insights.

Core Application Layer (Main GUI Controller)

- This layer acts as the primary "command center," controlling the entire analytical workflow and synchronizing all sub-modules.
- It manages the flow of user inputs and orchestrates the sequence of preprocessing, feature extraction, and model inference.
- The controller ensures smooth data communication between the underlying CSV datasets, the trained models, and the final output displays.
- Its modular design allows for the isolation of specific tasks, ensuring the system remains stable during intensive computational processes.

Dataset Input (CSV File)

- The system accepts customer support telemetry in CSV format as the foundational input for the analysis.
- The dataset comprises a rich mix of textual and categorical attributes, such as raw ticket descriptions, historical priority levels, and resolution logs.
- This raw data serves as the primary source for identifying patterns in customer sentiment and support efficiency.
- Once ingested, the data is automatically routed to the preprocessing module for linguistic refinement.

Data Preprocessing (NLP Processing)

- Raw support text undergoes a multi-stage cleaning and normalization process using established NLP techniques.

- The pipeline includes standard operations such as case standardization (lowercasing), tokenization, stopword removal, and POS-based lemmatization.
- These steps strip away linguistic noise, transforming unstructured text into a clean, structured format optimized for deep learning.
- This module ensures that the subsequent feature extraction stage focuses only on the most semantically relevant words.

Feature Extraction (XLNet Embeddings)

- The refined text is converted into high-dimensional numerical feature vectors using XLNet-based transformer embeddings.
- Tokenized inputs are passed through the pretrained XLNet model to generate context-aware representations that capture the subtle nuances of support language.
- Mean pooling is applied to the transformer outputs to obtain fixed-length vectors that encapsulate the core semantic information of each ticket.
- This approach allows the system to understand the "meaning" behind a customer's query rather than just matching keywords.

Feature Storage (Model Cache)

- To optimize performance, extracted XLNet features are persisted using the joblib library.
- This caching mechanism eliminates the need to re-run the intensive transformer model during repeated experiments or model tuning.
- By storing these "frozen" feature vectors, the system significantly reduces the computational overhead of the training cycle.
- This ensures that the framework remains responsive even when working with large volumes of support tickets.

Existing Models (QDA, LDA, NC, SGD)

- The system establishes a performance baseline by providing the XLNet features to a diverse group of traditional machine learning models.
- QDA and LDA are used to explore quadratic and linear decision boundaries for class separation.
- NC provides a distance-based perspective, while SGD offers a scalable linear approach for large datasets.
- These models generate independent predictions for ticket priority and satisfaction, serving as a benchmark for the proposed hybrid.

Proposed Model (HGB Classifier)

- The HGB classifier serves as the primary engine for high-accuracy prediction.
- It builds an ensemble of decision trees iteratively, specifically optimized for the high-dimensional vectors produced by XLNet.
- The HGB architecture is engineered for speed and memory efficiency, handling large-scale support data with significantly better accuracy than traditional learners.
- This model effectively captures the complex, non-linear relationships between ticket text and support outcomes.

Model Training and Evaluation

- The dataset is split into training and testing subsets to facilitate a rigorous validation of the HGB and baseline models.
- The system computes a full suite of performance metrics, including Accuracy, Precision, Recall, and F1-score, for each predicted target.
- Confusion matrices and ROC curves are generated dynamically to provide deep insights into the model's ability to distinguish between different priority and satisfaction levels.
- This evaluative data is critical for ensuring the system provides reliable guidance for support desk decision-making.

Prediction Module

- The inference engine applies the learned knowledge to new, incoming support tickets in real-time.
- New data is automatically routed through the exact same NLP preprocessing and XLNet extraction pipeline to maintain feature consistency.
- The trained HGB model generates simultaneous predictions for all three target variables.
- These predicted outputs are then mapped back to their original human-readable labels (e.g., "High Priority") for display in the GUI.

Visualization and Reporting

- The framework translates abstract model data into intuitive graphical outputs, such as Word Clouds and frequency plots of common support issues.
- Comparative performance tables are generated to show the relative effectiveness of each model across different support metrics.
- These visualizations allow administrators to quickly grasp data trends and trust the model's underlying logic.
- High-fidelity charts are used to track how satisfaction levels vary across different ticket categories.

System Output & Adaptive Workflow

- The final output provides actionable intelligence, including predicted ticket priority, satisfaction ratings, and the likely resolution status.
- The system's Adaptive Workflow allows users to re-run any stage of the pipeline from preprocessing to training with new parameters or datasets.
- This modularity ensures the system can be continuously refined as customer language patterns and support priorities evolve over time.
- The result is a complete analytical lifecycle that turns raw support data into a streamlined, automated operational strategy.

4. Results and Discussion

Fig. 2 presents a visually appealing Word Cloud representation of the top 100 most frequent words extracted from the pre-processed customer support tickets. Prominent terms such as "issue", "please", "assist", "product", "request", "problem", "refund", and "technical" appear in larger fonts, effectively highlighting the dominant themes and recurring language patterns in customer complaints and inquiries.

Fig. 2: Visualization top 100 words using Word Cloud.

Fig. 3 illustrates a horizontal bar chart displaying the top 20 most frequent words in the dataset along with their respective counts. The word "issue" leads significantly with the highest frequency, followed by "please", "assist", "product", "request", and other common support-related terms, clearly demonstrating the most prevalent vocabulary used by customers when describing their problems.

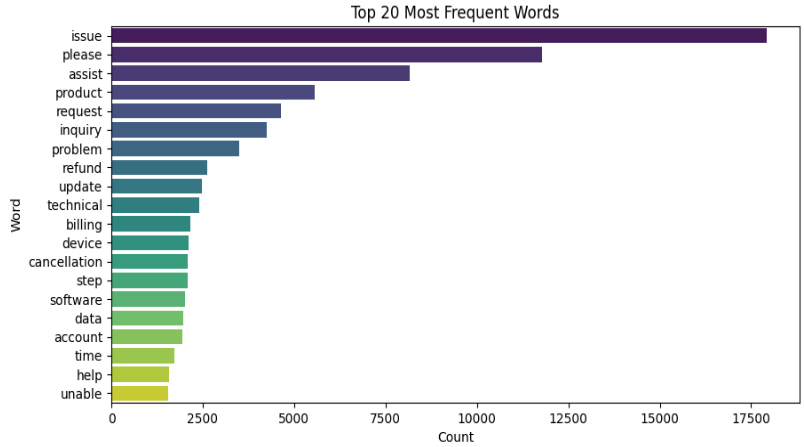

Fig. 3: Count plot of Top 20 Most Frequent Words.

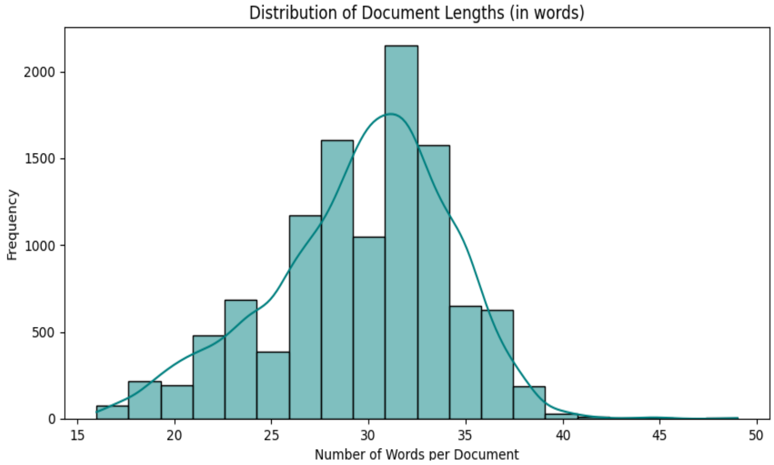

Fig. 4: Distribution of Document Lengths using Histogram.

Fig. 4 depicts a histogram showing the distribution of document lengths (in number of words) across all pre-processed support tickets. The distribution follows an approximately normal shape with a peak between 30–35 words per ticket and a right skew, indicating that most customer messages are relatively short and concise, typically containing 25–40 words.

Fig. 5 presents a bar chart illustrating the frequency distribution of different Parts of Speech (POS) tags in the tokenized text corpus. Nouns (NN) appear most frequently by a large margin, followed by adjectives (JJ) and verbs (VB forms), reflecting the descriptive and action-oriented nature of customer support tickets that frequently mention problems, products, and requested actions.

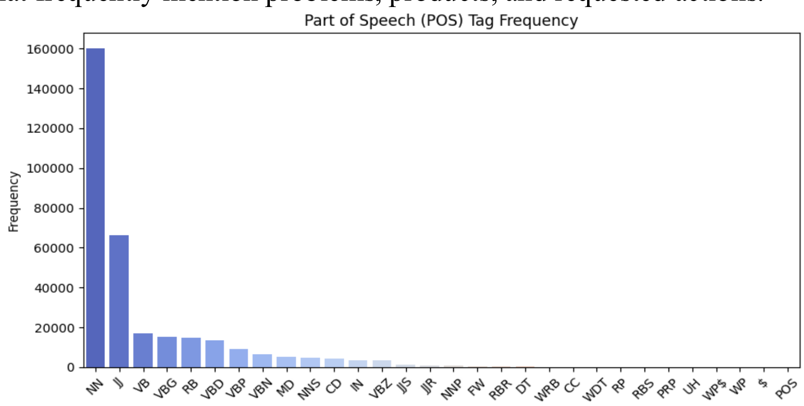

Fig. 5: Parts of Speech (POS) Tag Frequency.

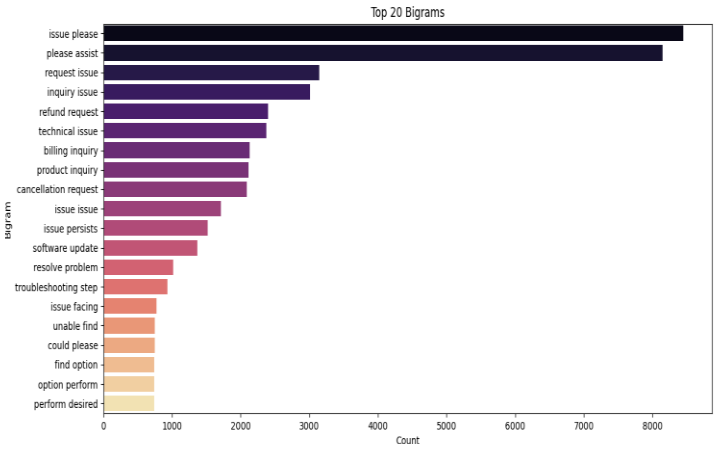

Fig. 6: Count plot of Top 20 Bigrams.

Fig. 6 displays a horizontal bar chart of the top 20 most common bigrams (two-word combinations) found in the support ticket texts. Phrases such as "issue please", "please assist", "request issue", "refund issue", and "inquiry request" dominate, clearly revealing typical polite request patterns and problem-reporting expressions frequently used by customers.

Figure 7 presents the confusion matrices for the "Ticket Priority" classification task using five different classifiers (proposed HGB) trained on XLNet embeddings. These matrices visually compare the models' prediction performance across four priority classes Critical, High, Low, and Medium by showing the distribution of true versus predicted labels on the test set. The diagonal elements represent correct predictions, while off-diagonal values indicate misclassifications, allowing a clear assessment of each model's strengths, weaknesses, and overall effectiveness in handling the multi-class ticket prioritization problem.

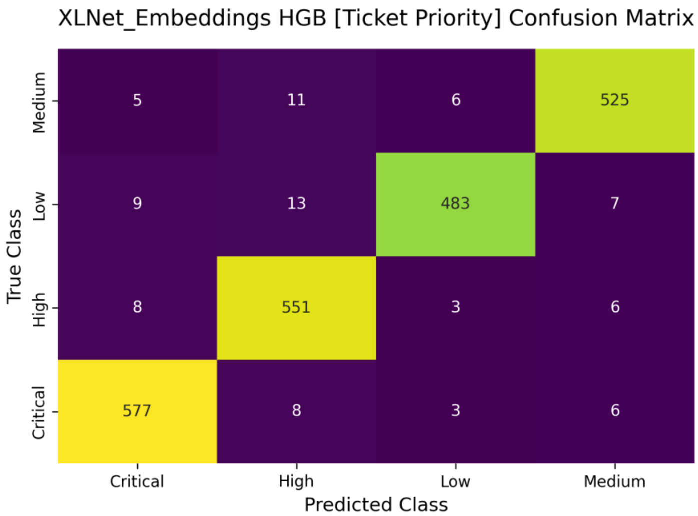

Fig. 7: Confusion Matrix obtained using HGB Classifier for column “Ticket Priority”.

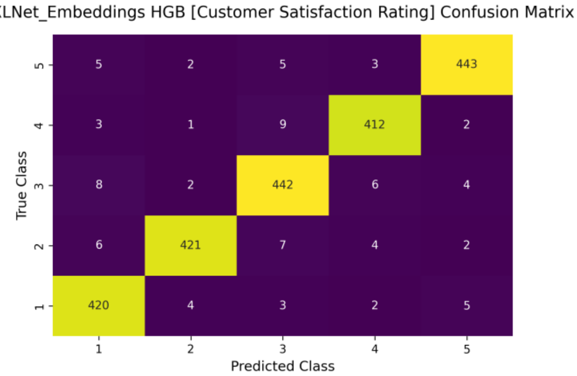

Fig. 8: Confusion Matrix obtained using HGB Classifier for column “Customer Satisfaction Rating”.

Fig. 8 HGB Classifier (Proposed Model) The HGB classifier produces an exceptionally strong confusion matrix with dominant diagonal elements across all five rating levels (e.g., 420/429 for rating 1, 421/440 for rating 2, 442/462 for rating 3, 412/426 for rating 4, and 443/458 for rating 5). Very few off-diagonal misclassifications occur, and most errors are limited to adjacent classes, demonstrating outstanding discriminative power and establishing HGB as the clearly superior model for predicting customer satisfaction ratings using XLNet embeddings.

Figure 9 presents the confusion matrices for the multi-class "Resolution" prediction task using five classifiers (QDA, NC, SGD, LDA, and the proposed HGB) trained on XLNet embeddings. This task involves classifying customer support tickets into one of several resolution categories (e.g., Software bug, Refund request, Product setup, Hardware issue, Network problem, etc.). The matrices show the distribution of true versus predicted resolution types, with rows representing actual classes and columns representing predicted classes. Diagonal values indicate correct predictions, while off-diagonal entries reveal misclassification patterns. The results highlight significant performance variation among the models, with the proposed HGB classifier demonstrating dramatically superior accuracy and much cleaner class separation compared to the baselines.

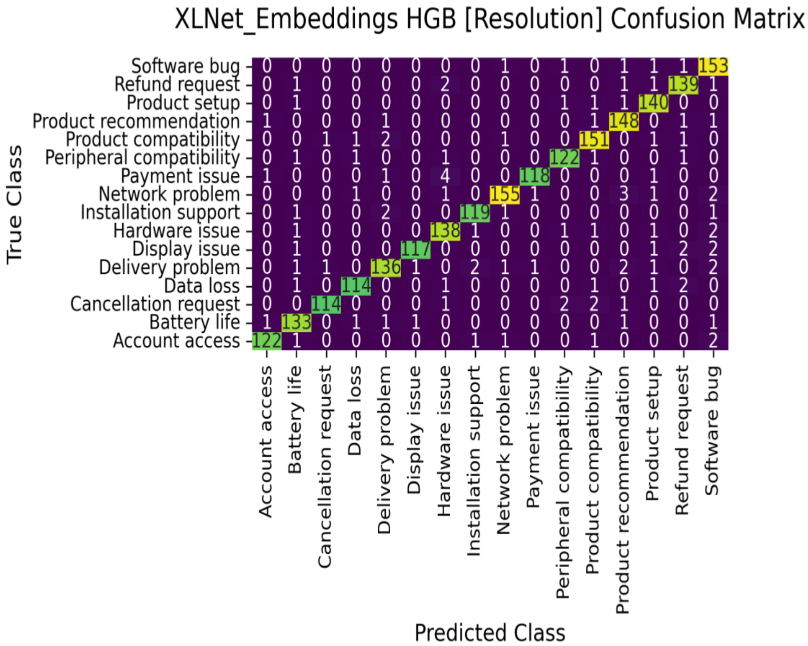

Fig. 9: Confusion Matrix obtained using HGB Classifier for column “Resolution”.

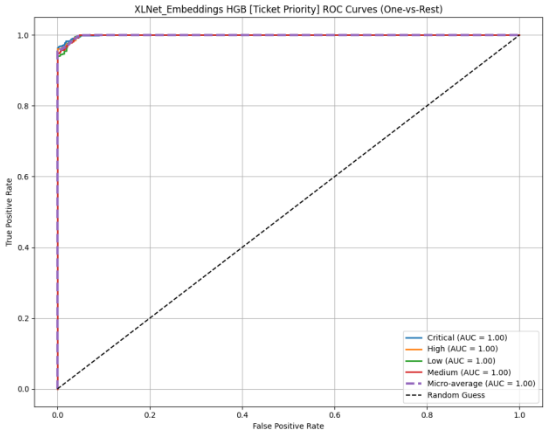

Fig. 10: Confusion Matrix obtained using HGB Classifier for column “Ticket Priority”.

Fig 10 illustrates the ROC curves and corresponding AUC values for the "Ticket Priority" classification task using four classifiers (QDA, NC, LDA, and the proposed HGB) trained on XLNet embeddings. The curves are plotted in the One-vs-Rest (OvR) fashion for each priority class (Critical, High, Low, Medium), along with the micro-average AUC and a random-guessing baseline (diagonal dashed line). Higher AUC values closer to 1.0 indicate better discriminative ability between classes. The figure clearly demonstrates a substantial performance progression from the baseline models to the proposed HGB classifier, which achieves near-perfect separation across all priority levels.

Figure 11 presents the ROC curves and corresponding AUC values for the multi-class "Customer Satisfaction Rating" prediction task (5 rating levels, from 1 to 5) using four classifiers (QDA, NC, LDA, and the proposed HGB) trained on XLNet embeddings. The curves are computed in the One-vs-Rest (OvR) manner for each rating class, along with the micro-average AUC and a random-guessing baseline (diagonal dashed line). Higher AUC values closer to 1.0 reflect stronger ability to discriminate between satisfaction levels. The figure clearly illustrates a marked performance improvement from the baseline models to the proposed HGB classifier, which achieves near-perfect classification across all rating categories.

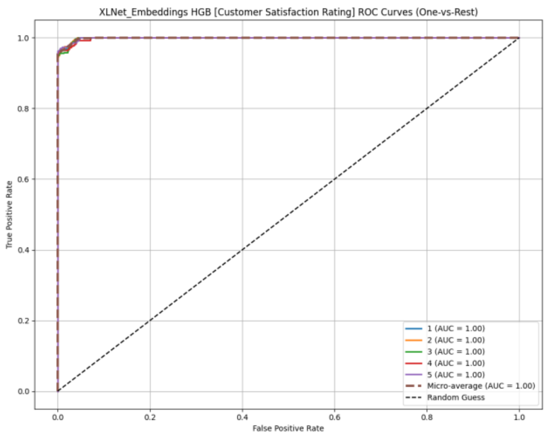

Fig. 11: Confusion Matrix obtained using HGB Classifier for column “Customer Satisfaction”.

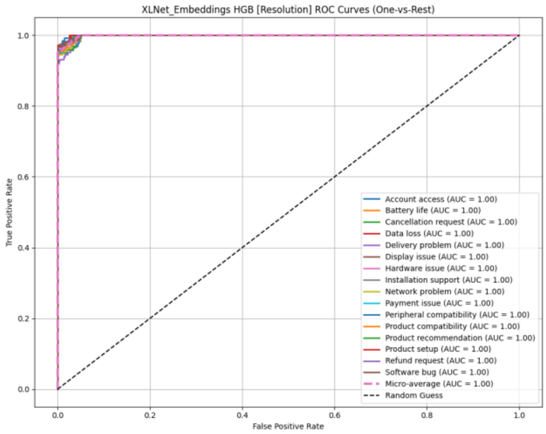

Fig. 12: Confusion Matrix obtained using HGB Classifier for column "Resolution".

Figure 12 displays the ROC curves and corresponding AUC values for the multi-class "Resolution" prediction task using four classifiers (QDA, NC, LDA, and the proposed HGB) trained on XLNet embeddings. The curves are generated in the One-vs-Rest (OvR) manner for each resolution category (e.g., Software bug, Refund request, Product setup, Hardware issue, Network problem, etc.), accompanied by the micro-average AUC and a random-guessing baseline (diagonal dashed line). Higher AUC values closer to 1.0 indicate superior ability to distinguish a given resolution type from all others. The figure clearly shows a dramatic performance progression, with baseline models exhibiting poor to moderate discrimination, while the proposed HGB classifier achieves near-perfect separation across virtually all resolution classes.

5. Conclusion

This study presents an effective and intelligent automated framework for predicting key attributes of customer support tickets, including priority level, customer satisfaction, and resolution outcomes. By integrating the contextual language understanding strengths of XLNet embeddings with reliable classical machine learning methods—especially the Histogram-Based Gradient Boosting classifier the approach achieves strong predictive performance. The hybrid design combines the advantages of transformer-based feature extraction with the efficiency and stability of traditional algorithms, leading to improved accuracy and scalability. It minimizes the need for manual processing, speeds up ticket handling and prioritization, and contributes to better overall customer service delivery. Furthermore, the inclusion of exploratory analysis and visualization techniques offers deeper insights into the patterns and characteristics of support ticket data. These insights help organizations refine their processes and continuously enhance the effectiveness of their customer support systems.

REFERENCES

[1] Aamodt, A., & Plaza, E. (1994). "Case-based reasoning: Foundational issues, methodological variations, and system approaches." AI Communications, 7(1), pp. 39-59.

[2] Alessandro Zangari, Matteo Marcuzzo, Michele Schiavinato, Andrea Gasparetto, Andrea Albarelli, Ticket automation: An insight into current research with appli cations to multi-level classification scenarios, Expert Systems with Applications, Volume 225, 2023, 119984, ISSN 0957-4174.

[3] Ali Zaidi, S.S., Fraz, M.M., Shahzad, M. and Khan, S., 2022. A multiapproach generalized framework for automated solution suggestion of support tickets. International Journal of Intelligent Systems, 37(6), pp.3654-3681.

[4] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., et al. (2020). "Language Models are Few-Shot Learners." arXiv preprint arXiv: 2005.14165.

[5] Cannon, D., &Wheeldon, D. (2007). ITIL Service Operation. The Stationery Office.

[6] Chinthapatla, Y. (May, 2024) "AI Integration With ServiceNow and CMDB: Revo lutionizing Industries and Society". Journal of Science & Technology, vol. 5, no. 3, pp. 1-13.

[7] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), pp. 4171-4186.

[8] Fensel, D., Facca, F. M., Simperl, E., & Toma, I. (2011). Semantic Web Services: Springer Science & Business Media. Conference Name: ACM Woodstock conference

[9] Gupta, H.S. and Sengupta, B., 2012. Scheduling service tickets in shared delivery. In Service-Oriented Computing: 10th International Conference, ICSOC 2012, Shanghai, China, November 12-15, 2012. Proceedings 10 (pp. 79-95). Springer Berlin Heidelberg.

[10] Huang,J.,Liu, X., King, I., &Lyu, M.R.(2015)."AnAdaptiveClusteringMethodfor Enhancing IT Service Management." IEEE Transactions on Services Computing, 8(6), pp. 894-908.

[11] Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). "Data Clustering: A Review." ACM Computing Surveys, 31(3), pp. 264-323.

[12] Lyubinets, V., Boiko, T.andNicholas, D., 2018, August.Automatedlabelingofbugs and tickets using attention-based mechanisms in recurrent neural networks. In 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP) (pp. 271-275). IEEE.

[13] Murtagh, F., & Contreras, P. (2012). "Algorithms for hierarchical clustering: an overview." Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2(1), pp. 86-97.

[14] Paramesh, S.P. and Shreedhara, K.S., 2019. Automated IT service desk systems using machine learning techniques. In Data Analytics and Learning: Proceedings of DAL 2018 (pp. 331-346). Springer Singapore.

[15] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). "Lan guage Models are Unsupervised Multitask Learners." OpenAI Blog, 1(8).

[16] Rajagopal, M., Ramkumar, S. (2023). Adopting Artificial Intelligence in ITIL for Information Security Management—Way Forward in Industry 4.0. In: Sarvesh waran, V., Chen, J.IZ., Pelusi, D. (eds) Artificial Intelligence and Cyber Security in Industry 4.0. Advanced Technologies and Societal Change. Springer, Singapore.

[17] Reinhard, P., Li, M.M., Dickhaut, E., Peters, C., Leimeister, J.M. (2023). Empower ing Recommender Systems in ITSM: A Pipeline Reference Model for AI-Based Textual Data Quality Enrichment. In: Gerber, A., Baskerville, R. (eds) Design Science Research for a New Society: Society 5.0. DESRIST 2023. Lecture Notes in Computer Science, vol 13873.

[18] Shitao Xiao and Zheng Liu and Peitian Zhang and Niklas Muennighoff (2023) C-Pack: Packaged Resources to Advance General Chinese Embedding. arXiv, 2309.07597 Vancouver