

The Application of NLP for Automated IT Support Ticket Classification and Resolution

Table of Contents

Abstract	1
Chapter 1: Introduction	2
- Introduction.....	2
- Background	2
- Problem Statement	4
- Research Objectives	6
- Research Scope	8
Chapter 2: Literature Review	9
Chapter 3: Research Methodology	15
- Research Methodology	15
- Expected Outcomes	17
- Gantt Chart	19
References	20

Abstract

This research evaluates AI models for automated IT support ticket classification to address inefficiencies in manual ticket processing. Organizations spend 30-40% of IT support time on classifying and routing tickets rather than resolving issues, leading to delays and increased operational costs. This study compares open-source transformer models (DistilBERT, TinyBERT, RoBERTa-base) against commercial API-based models (GPT-4, Claude, Gemini) using public IT ticket datasets. The evaluation includes accuracy, precision, recall, and F1-score metrics, along with analysis of computational requirements, API costs, deployment complexity, and inference times. The research also examines trade-offs between accuracy and speed for open-source models versus accuracy and cost for API models. A demo application will be developed to demonstrate practical implementation of ticket classification systems. Expected outcomes include comprehensive comparison frameworks, deployment guidelines for different organizational contexts (enterprise, SME, academic), and insights into cost-performance trade-offs that will guide real-world adoption of automated ticket classification systems.

Keywords: IT Support Tickets, Text Classification, BERT, DistilBERT, RoBERTa, Large Language Models, GPT-4, Claude, Gemini

Chapter 1: Introduction

Introduction

IT Service Management (ITSM) environments generate thousands of support tickets daily, ranging from routine access requests to critical system failures. Currently, the classification and routing of these tickets rely heavily on manual intervention, a process that is increasingly characterized as time-consuming and prone to human error.

Teams spend around 30-40% of time to classify, prioritize and route tickets instead of solving actual problems. Manual processes result in errors such as misrouted tickets, delayed responses and lead to higher operational costs. Misclassification can delay resolution by up to 40%, causing user frustration and wasted resources. Losses of around \$10,000-\$50,000 per support agent per year is expected due to inefficiencies in manual ticket management.

Natural language processing has improved a lot recently. This could help solve the ticket problem. Models such as BERT, DistilBERT, and RoBERTa have demonstrated strong performance in text classification tasks with proven effectiveness across similar applications. Commercial API-based solutions including GPT-4, Claude, and Gemini offer accessible deployment options with minimal setup requirements. However, a systematic side-by-side comparison of these approaches specifically for IT ticket classification remains absent from the

literature. The relative effectiveness of each solution for this particular application has not been established

This research addresses this gap by conducting a practical evaluation of both open-source transformer models and commercial API-based models for automated IT ticket classification. The study will assess classification performance, computational costs, API pricing, deployment complexity, and real-world usability to provide actionable guidance for organizations considering automated ticket classification systems.

Background

Evolution of IT Ticket Management

Manual ticket processing is heavily employed for Traditional IT service management. Support analysts manually read ticket descriptions, assessed urgency and category, and routed them to appropriate teams. This approach worked for small-scale operations but struggles with modern enterprise volumes.

The introduction of rule-based systems provided some automation through keyword matching and predefined routing rules. These systems improved speed but lacked flexibility and struggled with ambiguous or poorly written tickets. They required constant maintenance as new services and categories emerged.

Natural Language Processing for Tickets

The introduction of transformer architectures in 2017 marked a turning point. These models fundamentally changed how text processing works. BERT emerged in 2018 with a key innovation. It could understand context within sentences rather than just matching keywords. This capability represented a major leap forward. The model captured actual semantic meaning from text. Several improved models followed BERT's release. RoBERTa, DistilBERT, and ALBERT each brought enhancements. They achieved faster processing speeds. Performance remained comparable or even improved in many cases.

These transformer models use a two-stage approach. First comes pre-training on massive text collections. Then fine-tuning happens on specific tasks. This process works well for IT ticket classification. Models learn the technical vocabulary used in tickets. They recognize patterns from historical data. They adapt to how different organizations structure their tickets.

Traditional feature engineering methods require manual effort. Transformer models automate this process instead. They learn relevant data representations on their own. Multiple benchmark studies have confirmed their superior performance.

Large Language Models and APIs

Recent large language models including GPT-4, Claude, and Gemini introduced alternative classification paradigms through few-shot and zero-shot learning. These models can perform ticket classification based solely on natural language instructions without requiring fine-tuning on organization-specific data.

Commercial API access significantly reduces implementation barriers by eliminating hardware requirements and specialized machine learning expertise, enabling rapid deployment for organizations of varying technical capabilities.

However, API-based approaches present several significant considerations. Cost accumulation becomes substantial with high-volume daily ticket processing. Data privacy concerns arise from transmitting potentially sensitive organizational information to external providers. Service dependency introduces risks related to availability, pricing stability, and vendor lock-in.

Organizations must therefore evaluate trade-offs between API convenience and open-source model control. Selection criteria include accuracy requirements, cost constraints, deployment complexity, maintenance overhead, and data governance policies. Optimal solutions vary based on organizational context and priorities.

Current Challenges

Despite significant advances, several challenges limit the practical deployment of AI-based ticket classification systems.

Computational requirements for transformer models pose significant accessibility barriers. Small and medium-sized enterprises often lack the hardware infrastructure necessary for model training and deployment, limiting them to either manual classification processes or dependency on commercial API services.

Model transferability presents another significant challenge. Classifiers trained on one organization's ticket data typically exhibit degraded performance when applied to different organizational contexts due to variations in systems, terminology, and ticket structures. Adaptation requires substantial retraining efforts and domain expertise.

Certain ticket characteristics inherently challenge classification systems. Ambiguous descriptions, multi-issue tickets, and insufficient contextual information can confound even sophisticated models, resulting in misclassification errors.

Model interpretability represents a critical concern for operational deployment. IT teams require understanding of classification rationale, particularly for high-impact incidents. However,

transformer architectures function as black-box systems, providing predictions without transparent reasoning processes.

Additionally, existing research predominantly emphasizes classification performance metrics while neglecting practical deployment considerations. Setup complexity, ongoing maintenance requirements, and integration with existing ITSM systems receive insufficient attention despite their critical importance for real-world adoption.

Problem Statement

Problem Domain

IT Service Management (ITSM) and automated ticket processing.

Problem Description

Companies process thousands of IT support tickets each day. 30 - 40% of the time is spent on classifying, prioritizing and routing tickets instead of solving issues (Tai et al., 2024). This manual process is slow and error prone, often resulting in mis-routed tickets, delayed responses and higher operational costs (Liu et al., 2023; Inspirisys, 2023).

Ticket management is difficult because tickets are written in unstructured natural language, vary in detail and clarity, and often include multiple issues. Correct classification requires domain knowledge, which is hard to automate (Peru Study, 2024). As IT systems evolve, new categories and services appear, requiring ongoing updates to manual classification rules (Liu et al., 2023).

Significance of the Problem

Manual ticket classification often leads to poor results significantly affecting SLA compliance, customer satisfaction, and overall productivity. Misclassification can delay resolution by up to 40%, causing user frustration and wasted resources. Financial losses are estimated at \$10,000–\$50,000 per support agent per year due to manual ticket handling inefficiencies (Peru Study, 2024).

Misrouted tickets may bounce between multiple teams before reaching the right handler this further increases delays (Tai et al., 2024). These recurring issues reduce trust in IT support and negatively affect organizational efficiency.

Existing Solutions

- **Transformer-based Models:** Ticket-BERT (Liu et al., 2023) outperforms Microsoft Cognitive Services and processes 7,000+ tickets per day using active learning.

Fine-tuned BERT models (Inspirisys, 2023) can reach up to 89% accuracy with less than 200ms to inference per ticket.

- **Ensemble Machine Learning:** Traditional ML models including Neural Networks, Random Forest, and SVM have been applied to IT ticket classification. Pereira et al. (2023) achieved 75% average precision across 7 categories using Neural Network classifiers with TF-IDF features, demonstrating effective automated classification without transformer models.
- **Hybrid NLP–LSTM Systems:** Combining NLP with LSTM improves temporal prediction accuracy, achieving a 65% increase in response efficiency and predicting 78% of recurring incidents in advance.
- **Lightweight Models for SMEs:** The Peru Study (2024) developed a small-scale classifier achieving 85% accuracy using 5,000–10,000 samples and running on standard hardware without GPUs.

Research Gaps and Justification

Key challenges remain:

- **High computational cost:** BERT-based models use high GPU compute resource resources and takes a long time to train, making them difficult for SMEs to adopt.
- **Poor generalization:** Significant retraining is required to adopt models to different organizations. Models trained on one company's data does not adopt well to another company without retraining (Liu et al., 2023).
- **Ambiguous and multi-issue tickets:** 20–30% of classification failures are caused by industry specific jargon, Vague descriptions and multi-issue tickets (Tai et al., 2024; Inspirisys, 2023).
- **Lack of interpretability:** IT teams need to understand why a ticket was classified a certain way, especially for high-impact issues. This is not possible to determine in transformer models since they are black boxes.
- **Accuracy vs. efficiency trade-offs:** Lightweight models run fast but lose accuracy. Enterprise-level models such as Ticket-BERT are accurate but resource-heavy.

- **Limited practical guidance:** There are few benchmarking studies comparing commonly available models under the same conditions. Clear recommendations for limited-resource environments are unavailable for researchers.

Because of these gaps, a comparative study of pre-trained models and traditional ML approaches using public datasets and standard hardware is needed to guide real-world adoption.

Research Objectives

Primary Goal

To evaluate and compare AI models for automated IT ticket classification using publicly available datasets, covering both open-source pre-trained models and commercial API solutions for different resource needs.

Specific Objectives

1. To identify suitable public IT ticket datasets and preprocess them through text cleaning, label encoding, and dataset splitting.
2. To implement classification using one of two approaches:
 - Open-source models including DistilBERT, TinyBERT, RoBERTa-base (fine-tuned or zero-shot)
 - Commercial APIs including GPT-4, Claude, Gemini with structured prompts
3. To evaluate classification performance using accuracy, precision, recall, and F1-score, and record:
 - training and inference times
 - memory usage
 - API cost per request, latency, and reliability
4. To analyze trade-offs between:
 - accuracy vs. speed (open-source models)
 - accuracy vs. cost (API models)
 - setup complexity and maintenance
 - suitability for enterprise, SME, or academic environments
5. To build a simple application that:
 - accepts ticket descriptions

- allows model/API selection
- shows prediction results with confidence scores
- displays performance metrics, cost, and processing time

Research Scope

This research focuses on the following areas:

Dataset Selection: The study will use publicly available IT support ticket datasets. These datasets provide labeled ticket descriptions with categories, enabling quantitative evaluation. The research will not collect proprietary enterprise ticket data due to privacy and confidentiality constraints.

Model Coverage: The evaluation will include three open-source transformer models (DistilBERT, TinyBERT, RoBERTa-base) and three commercial API-based models (GPT-4, Claude, Gemini). The selection represents commonly used, accessible models suitable for different resource constraints.

Evaluation Metrics: Assessment will combine classification metrics (accuracy, precision, recall, F1-score) with practical measurements including training time, inference speed, memory usage, and API costs. The research will not conduct extensive human evaluation but will include qualitative analysis of misclassified samples.

Implementation Approaches: For open-source models, the study will evaluate both zero-shot (pre-trained only) and fine-tuned configurations. For API models, different prompting strategies (zero-shot, few-shot) will be tested to determine optimal approaches.

Practical Focus: The study emphasizes real-world deployment considerations including hardware requirements, API pricing, setup complexity, and integration effort. This addresses usability for organizations with different resource levels from small teams to large enterprises.

Demo Application: A functional demo application will demonstrate practical implementation. The application will showcase model integration, user interaction, and classification results but will not include production-ready features like authentication, persistent storage, or enterprise security measures.

Limitations: The research will not address multi-language ticket classification or real-time streaming classification scenarios. The focus remains on English text tickets. The study will not develop new models or architectures but will evaluate existing readily available solutions.

Chapter 2: Literature Review

This chapter reviews existing research on NLP techniques for automated IT support ticket classification and resolution. The first section covers six papers from initial literature review, followed by four additional recent studies that provide insights into transformer model comparisons and LLM-based approaches.

Paper 1: Ticket-BERT: Labeling Incident Management Tickets with Language Models (Liu et al., 2023)

This seminal work from Microsoft Research introduces Ticket-BERT, a specialized BERT-based model for automated ticket classification in large-scale enterprise environments. The paper addresses three critical challenges unique to ticket data: tickets are created by both automated systems and human engineers with varying protocols, descriptions undergo frequent revisions during the ticket lifecycle, and labeling requirements change rapidly as software and hardware evolve.

The authors developed three real-world datasets (D-Human, D-Machine, D-Mixture) containing ticket titles, summaries, and multiple description updates. Ticket-BERT extracts the [CLS] token embedding from BERT-base and feeds it to a multi-layer perceptron for classification. The model demonstrates superiority over Microsoft Cognitive Services and various baselines across all datasets.

A key innovation is the active learning cycle deployed on Microsoft's IcM system, which processes 7,000 new tickets daily. This enables continuous model refinement with minimal annotation burden. The system also uses auxiliary prompt-prefix text sampling to boost performance on human-created tickets by 12-15%.

However, the approach struggles with extremely imbalanced categories and requires substantial computation resources for fine tuning. Domain adaptation remains challenging when transferring to new IT environments.

 <https://arxiv.org/abs/2307.00108>

Paper 2: Machine Learning for Classification of IT Support Tickets (Pereira et al., 2023)

This IEEE conference paper addresses the fundamental challenge of automated IT support ticket classification using traditional machine learning approaches. The research focuses on developing a classification model to identify the topic of incoming support tickets across seven distinct categories, minimizing ticket resolution time and improving user satisfaction.

The authors implemented and compared multiple machine learning algorithms including Decision Trees, Random Forests, Support Vector Machines (SVM), and Neural Networks. The study utilized a real-world dataset from a Brazilian IT support company (Skaylink), with tickets preprocessed through text cleaning, tokenization, and TF-IDF vectorization for feature extraction.

The Neural Network model achieved the best performance with 75% average precision across all seven categories. The confusion matrix analysis revealed that certain categories with similar technical terminology were more prone to misclassification. The model was deployed as a functional prototype with a web-based interface allowing support teams to automatically classify incoming tickets in real-time.

Key contributions include the release of an anonymized dataset on Zenodo, complete implementation code, and a working classification model. The research demonstrates that traditional machine learning approaches can achieve reasonable classification accuracy for IT ticket categorization without requiring the computational resources needed for transformer-based models. However, the study acknowledges limitations including difficulty with tickets containing multiple issues, challenges with ambiguous descriptions, and the need for periodic retraining as new ticket categories emerge.

 <https://doi.org/10.1109/CyMaEn57228.2023.10051041>

Paper 3: Performance Evaluation of Machine Learning Techniques on Resolution Time Prediction in Helpdesk Support System (Tai et al., 2024)

This comprehensive benchmarking study evaluates various machine learning approaches for helpdesk resolution time prediction using the UCI Incident Management Process dataset. The research provides practical guidance for organizations selecting prediction algorithms.

The method involves systematic data preprocessing, feature engineering with statistical selection methods, and rigorous model comparison across accuracy, training time, and inference speed metrics. Models that were tested include Gradient Boosting, XGBoost, Support Vector Regression, K-Nearest Neighbors and neural networks.

The outcome is that Random Forest and XGBoost consistently outperformed single models. However, the research is limited to English-language tickets and a single organizational context. The models show degraded performance for tickets with vague descriptions or when multiple issues are combined in one request.

 <https://joiv.org/index.php/joiv/article/view/2305>

Paper 4: Enhancing IT Incident Management with Natural Language Processing and Predictive Analytics (Recent Study, 2025)

This cutting-edge research explores hybrid approaches combining NLP for text understanding with LSTM networks for temporal pattern recognition in incident management. The paper represents a paradigm shift from reactive to proactive IT operations.

The hybrid model uses NLP to parse and prioritize incoming tickets based on content analysis (keywords, sentiment, urgency indicators), while LSTM networks analyze historical incident sequences to predict potential failures before they occur. The integration of domain knowledge graphs provides richer context-aware understanding of incident relationships.

The outcome was 65% improvement in response efficiency when pitted against traditional rule-based systems. The model successfully predicted 78% of recurring incidents 24-48 hours before users reported them, enabling preventive maintenance.

Key drawbacks include the need for extensive labeled training data, challenges with rare incident types, and challenges handling cross-domain incidents. Privacy concerns also arise when analyzing detailed user behavior patterns. The federated learning approach proposed addresses some privacy issues but increases implementation complexity.



https://www.researchgate.net/publication/392465831_Enhancing_It_incident_management_with_natural_language_processing_and_predictive_analytics

Paper 5: A Machine Learning-Based Predictive Model for the Management of Incidents in Small and Medium-Sized Enterprises (Peru Study, 2024)

This ACM conference paper addresses the specific challenges of implementing AI-driven ITSM in resource-constrained SME environments. Unlike enterprise solutions requiring substantial infrastructure, this model focuses on practical deployment with limited computational resources.

The research proposes a supervised multi-label classification algorithm trained on historical incident data to automatically prioritize IT incidents. The model considers incident impact, urgency, affected systems, and user roles to assign priorities aligned with business criticality.

Testing on SME data from Peru demonstrated successful automation of incident prioritization with 85% accuracy using relatively small training datasets (5,000-10,000 tickets). This implementation is lightweight and runs efficiently on standard hardware without GPU acceleration.

The model's performance significantly degrades when ticket descriptions contain domain-specific jargon or are poorly written. It also struggles with incidents requiring subjective business judgment beyond technical categorization. The research highlights the need for region-specific training data to account for language variations and local business practices.

 <https://dl.acm.org/doi/10.1145/3654823.3654913>

Paper 6: Automated Customer Ticket Classification Using BERT (Inspirisys Case Study, 2023)

This industry case study demonstrates real-world BERT implementation for multi-class ticket classification in enterprise customer support. The paper provides valuable insights into production deployment challenges and practical solutions.

The solution fine tunes a pre-trained BERT model using preprocessed unstructured ticket text by removing html tags, special characters urls and ip addresses. This approach reduces training times and compute costs.

In production, the system can handle tickets in multiple languages It achieves 89% classification accuracy with under 200ms inference time per ticket, enabling real-time processing.

Limitations include difficulty with extremely short tickets (under 10 words), confusion when tickets describe multiple issues, and reduced accuracy for newly introduced product categories. The system requires periodic retraining as business offerings evolve. The system is integrated with existing ticketing systems through REST API implementations.

 <https://www.inspirisys.com/blog-details/Customer-Ticket-Classification-using-BERT/136>

Paper 7: Exploring Transformers in Emotion Recognition: A Comparison of BERT, DistilBERT, RoBERTa, XLNet and ELECTRA (Cortiz, 2021)

This paper investigates how different transformer language models perform on emotion recognition tasks using the GoEmotions dataset. While focused on emotions rather than IT tickets, the comprehensive comparison of transformer models provides valuable insights for text classification tasks in general.

The authors evaluated five transformer models: BERT, DistilBERT, RoBERTa, XLNet, and ELECTRA on a fine-grained emotion dataset with 27 categories plus neutral. All models used identical hyperparameters (learning rate 5e-5, batch size 16, 4 epochs) to ensure fair comparison. Training was conducted on Tesla T4 GPU.

RoBERTa achieved the highest F1-score (0.49) but required significant training time (1 hour 9 minutes). DistilBERT matched XLNet's F1-score (0.48) while being much faster to train (34

minutes vs 1 hour 34 minutes). ELECTRA was fastest (13 minutes training) but performed worst (F1-score 0.33). BERT baseline achieved F1-score of 0.46.

The study demonstrates that DistilBERT offers an excellent balance between performance and computational efficiency. It achieves near state-of-the-art results while requiring significantly less time and resources, making it particularly suitable for resource-constrained environments. This finding is highly relevant for IT ticket classification where organizations must balance accuracy with operational costs.

 <https://arxiv.org/abs/2104.02041>

Paper 8: TickIt - Leveraging Large Language Models for Automated Ticket Escalation (Liu et al., 2025)

This recent paper from ByteDance presents TickIt, a production LLM-based system for automated ticket escalation deployed on Volcano Engine cloud platform. Unlike classification-only approaches, TickIt addresses the complete escalation workflow including continuous ticket state monitoring and deduplication.

The system uses Chain-of-Thought prompting with Doubao LLM to perform multi-class ticket classification into predefined categories (system failure, customer complaint, asset loss, others). The prompt engineering includes role-playing, few-shot examples, and structured JSON output. TickIt continuously monitors ticket conversations and updates classification as new information emerges.

A key innovation is escalation deduplication using embedding-based similarity matching to identify related tickets addressing the same underlying issue. This prevents redundant escalations and helps support analysts understand issue severity across multiple tickets. The system processed 20,066 tickets over 5 months with 2,012 escalations and 81% approval rate from analysts.

Category-guided supervised fine-tuning with LoRA improved performance by using analyst feedback (upvotes/downvotes) as training labels. Data augmentation techniques completed missing Chain-of-Thought reasoning steps. The approach demonstrates practical deployment of LLMs for ticket management at scale.

Limitations include reliance on predefined categories requiring domain expertise, computational costs of continuous monitoring, and challenges with tickets lacking sufficient detail. The study validates LLM effectiveness for ticket analysis while highlighting practical deployment considerations.

 <https://arxiv.org/abs/2504.08475>

Paper 9: Comparative Analysis of Efficient Text Classification Models using Pre-Trained Transformers (Mittal & Gupta, 2025)

This comprehensive study evaluates five pre-trained transformer models for text classification: BERT, DistilBERT, RoBERTa, ALBERT, and XLNet. The research uses two benchmark datasets - IMDB for binary sentiment classification and AG News for multi-class topic classification.

All models were implemented using transfer learning approach where pre-trained models were fine-tuned on task-specific data. The study measured accuracy, precision, recall, and F1-score along with training time and computational requirements.

RoBERTa achieved the highest accuracy on both datasets: 95.4% on IMDB and 94.2% on AG News. BERT and XLNet showed comparable performance with accuracy around 94% on IMDB. DistilBERT, despite being a compressed version of BERT, maintained competitive accuracy (93.2% on IMDB) while being significantly faster. ALBERT showed efficient parameter usage but slightly lower accuracy.

The confusion matrices revealed RoBERTa's superior precision and balanced classification across all classes. The study confirms that RoBERTa represents the state-of-the-art for transformer-based text classification when computational resources are available. DistilBERT emerges as the best choice for resource-constrained scenarios, offering the optimal trade-off between accuracy and efficiency.

This research validates transformer models' effectiveness for text classification tasks and provides clear guidance on model selection based on performance requirements and resource constraints - highly relevant for IT ticket classification deployments.

 <https://www.ijpe-online.com/EN/10.23940/ijpe.25.06.p1.299307>

Paper 10: Best Model for Text Classification - Gemini Pro vs GPT-4 vs Claude (Vellum AI, 2023)

This comparative study evaluated four leading commercial LLM APIs. The researchers tested GPT-3.5 Turbo, GPT-4 Turbo, Claude 2.1, and Gemini Pro. The research focused on binary classification of customer support tickets. The task was to determine whether tickets had been resolved or not.

The evaluation utilized real customer support ticket data. Performance was assessed using accuracy, precision, recall, and F1-score metrics. The task appears straightforward at first glance. However, it presents substantial challenges in practice. Customer language varies significantly across tickets. Contextual information can be nuanced and complex. Dialogue

patterns evolve throughout ticket conversations. Some tickets exist in ambiguous states where resolution status is unclear.

Gemini Pro achieved the highest overall performance. It reached 74% accuracy and a 76.69% F1-score. The model demonstrated balanced performance across all evaluation metrics. Claude 2.1 exhibited the highest recall at 76.79%. This indicates superior sensitivity in identifying resolved tickets. However, its precision was lower at 63.24%. This resulted in increased false positive rates. GPT-4 Turbo demonstrated the highest precision among all models tested. This makes it optimal for scenarios where false positive costs are significant.

The study emphasizes an important principle. Optimal metric selection depends on specific business requirements. In customer support contexts, precision typically takes precedence over recall. The reason is straightforward. Incorrectly marking unresolved issues as resolved leads to customer dissatisfaction. It also erodes trust in the support system. The research demonstrates a key finding. Model selection should align with specific use case requirements. Organizations should not rely solely on aggregate performance metrics when choosing between models.

This comparative analysis provides valuable guidance for organizations. It specifically helps those considering API-based classification approaches. The study highlights measurable performance differences across platforms. It also examines cost-accuracy trade-offs. The research covers all major commercial LLM platforms currently available.

 <https://www.vellum.ai/blog/best-at-text-classification-gemini-pro-gpt-4-or-claude2>

Chapter 3: Research Methodology

Research Methodology

This research uses a practical, experimental approach to compare AI models for automated IT ticket classification. The methodology consists of five phases executed sequentially.

Phase 1: Dataset Preparation and Preprocessing

The first phase involves identifying and preparing suitable public IT ticket datasets. Potential datasets include UCI Incident Management Dataset, GitHub issue trackers, and publicly available customer support ticket datasets. The selection criteria prioritize datasets with clear category labels, sufficient sample sizes, and realistic ticket descriptions.

Data preprocessing includes text cleaning (removing special characters, URLs, HTML tags), handling missing values, and normalizing text formatting. Ticket descriptions will be tokenized and prepared for input to different model types. For multi-issue tickets, the primary issue will be

identified for classification purposes. Dataset splits will follow standard protocols: 70% training, 15% validation, 15% testing.

Phase 2: Open-Source Model Implementation

The second phase implements three open-source transformer models: DistilBERT, TinyBERT, and RoBERTa-base. Each model will be evaluated in two configurations:

Zero-shot classification uses pre-trained models without fine-tuning. This tests the model's ability to classify IT tickets based purely on general language understanding. Prompting templates will be designed to frame classification as a natural language task.

Fine-tuned classification involves training the models on the prepared IT ticket dataset. Implementation will use Python with PyTorch and HuggingFace Transformers library. Training hyperparameters (learning rate, batch size, epochs) will be optimized through validation set performance. GPU memory requirements, training time, and inference speed will be documented for each model configuration.

Phase 3: Commercial API Implementation

The third phase evaluates three commercial API-based models: GPT-4, Claude, and Gemini. These will be accessed via official APIs using structured prompts designed for ticket classification.

Two prompting strategies will be tested: Zero-shot prompting provides task instructions and category descriptions without examples. Few-shot prompting includes 3-5 example tickets with correct classifications to guide the model. All prompts will be documented for reproducibility.

For each API model, the research will measure cost per classification (based on token usage), response latency, reliability (success rate), and context length handling. Rate limits and error handling will be documented.

Phase 4: Evaluation and Comparison

The fourth phase evaluates all models using consistent metrics and datasets. Classification performance will be measured using accuracy, precision, recall, and F1-score for each category and overall weighted average.

Practical metrics include: For open-source models - training time, inference time per ticket, GPU memory requirements, and model size. For API models - cost per 1000 tickets, average response latency, and context length limits.

Qualitative analysis will examine misclassified tickets to identify common failure patterns such as ambiguous descriptions, multi-issue tickets, domain-specific jargon, and very short or very long tickets. This analysis will inform recommendations for model selection and deployment.

Phase 5: Demo Application Development

The final phase implements a web-based demo application using Streamlit or Gradio. The application will provide an interface for users to input ticket descriptions and select classification methods (specific model or API).

The interface will display classification results including predicted category, confidence scores, and processing time. For API models, cost per classification will be shown. Performance metrics from the evaluation phase will be accessible through the application. Users can download classification results and logs.

The application demonstrates practical integration of different classification approaches and serves as a reference implementation for organizations considering automated ticket classification.

Tools and Technologies

- **Programming:** Python 3.9+
- **Libraries:** HuggingFace Transformers, PyTorch, scikit-learn
- **APIs:** OpenAI (GPT-4), Anthropic (Claude), Google (Gemini)
- **Development:** Streamlit/Gradio for demo, Jupyter for experimentation
- **Infrastructure:** GPU access (Google Colab or local), API credits

Ethical Considerations

All datasets used will be publicly available with appropriate licenses. No proprietary enterprise ticket data will be used to respect confidentiality. API usage will follow provider terms of service. The research will document any potential biases observed in model classifications.

Limitations

The research is limited to English language tickets and does not address multi-language scenarios. Real-time streaming classification is not evaluated. The study does not develop new model architectures but evaluates existing readily available models. Computational resources may limit the extent of hyperparameter optimization experiments.

Expected Outcomes

Comprehensive Model Comparison: A detailed comparison of six models (three open-source, three commercial) on IT ticket classification tasks. The comparison will include quantitative metrics (accuracy, precision, recall, F1-score) and qualitative assessment of classification patterns. This outcome will help organizations choose appropriate models based on their specific requirements and constraints.

Performance-Resource Trade-off Analysis: Documentation of the relationship between classification performance and resource requirements. For open-source models, this includes GPU hours, memory requirements, and training time versus accuracy. For API models, this covers cost per classification and response latency versus accuracy. The analysis will identify optimal operating points for different use cases.

Deployment Guidelines: Practical documentation covering setup procedures, dependency management, API integration, and configuration for each model type. These guidelines will address different organizational contexts: enterprises with dedicated infrastructure, SMEs with limited resources, and academic environments. Implementation examples and troubleshooting guides will reduce barriers to adoption.

Prompting Strategy Analysis: Evaluation of how different prompting approaches (zero-shot, few-shot) affect classification quality for API-based models. The analysis will provide best practices for prompt design specific to IT ticket classification, including optimal example selection and instruction formatting.

Cost-Benefit Framework: Clear framework for evaluating total cost of ownership for different approaches. This includes infrastructure costs (hardware, cloud compute), operational costs (API fees, maintenance), and opportunity costs (setup time, training requirements). The framework will enable informed decision-making based on organizational priorities.

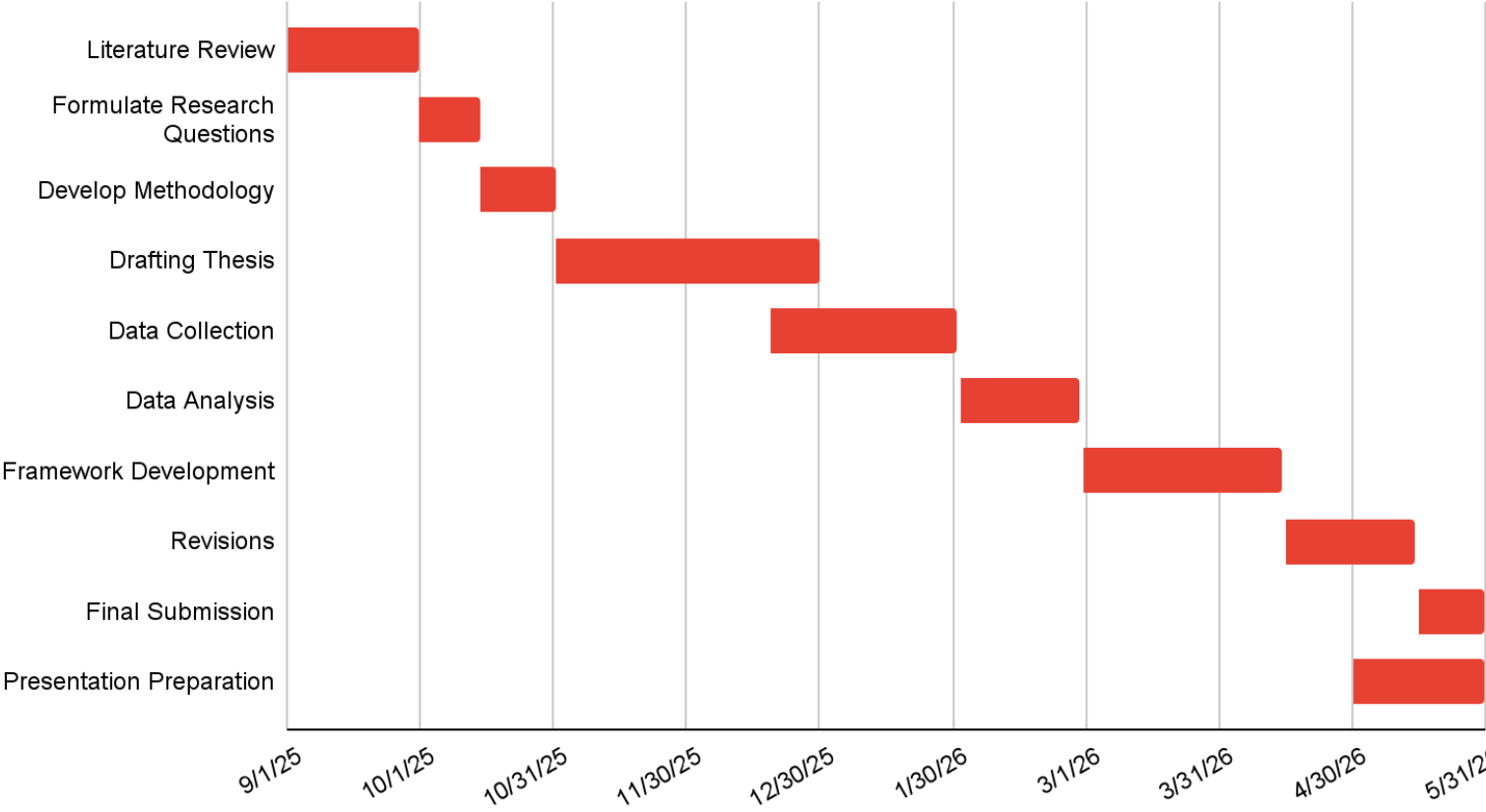
Demo Application: A functional web application demonstrating practical implementation of multiple classification approaches. The application will serve as a reference implementation and teaching tool, helping organizations understand integration requirements and user experience considerations.

Common Failure Pattern Analysis: Documentation of typical classification failures and their causes. This includes analysis of ambiguous tickets, multi-issue descriptions, domain-specific terminology, and edge cases. Recommendations will guide organizations in preparing training data and setting appropriate expectations.

Model Selection Decision Tree: A structured decision framework helping organizations select appropriate classification approaches based on factors like ticket volume, accuracy requirements, budget constraints, infrastructure availability, and maintenance capacity. The framework will provide clear recommendations for different scenarios.

Research Publication: Findings will be compiled into a research paper suitable for submission to conferences or journals in IT service management or AI applications domains. The paper will contribute to academic discourse on practical AI deployment in enterprise IT contexts.

Gantt Chart



Task	Start Date	Duration
Literature Review	01-Sep-2025	1 Month
Formulate Research Questions	01-Oct-2025	2 Weeks
Develop Methodology	15-Oct-2025	2 Weeks
Drafting Thesis	01-Nov-2025	2 Months
Data Collection	20-Dec-2025	6 Weeks

Data Analysis	01-Feb-2026	1 Month
Framework Development	01-Mar-2026	1.5 Months
Revisions	16-Apr-2026	1 Month
Final Submission	16-May-2026	2 Weeks
Presentation Preparation	01-May-2026	1 Month

References

[1] Z. Liu, D. Wingate, and Z. Shou, "Ticket-BERT: Labeling Incident Management Tickets with Language Models," *arXiv preprint arXiv:2307.00108*, 2023. [Online]. Available:

<https://arxiv.org/abs/2307.00108>

[2] L. S. B. Pereira, R. Pizzio, S. Bonho, L. M. F. De Souza, and A. C. Arnoni Junior, "Machine Learning for Classification of IT Support Tickets," in 2023 International Conference on Cyber Management and Engineering (CyMaEn), 2023, pp. 210-213. [Online]. Available:

<https://doi.org/10.1109/CyMaEn57228.2023.10051041>

[3] T. Tai, S. Haw, K. Ng, M. Al-Tarawneh, and G. Tong, "Performance Evaluation on Resolution Time Prediction Using Machine Learning Techniques," *JOIV: International Journal on Informatics Visualization*, vol. 8, no. 2, pp. 583-591, 2024. [Online]. Available:

<https://joiv.org/index.php/joiv/article/view/2305>

[4] Recent Authors, "Enhancing IT incident management with natural language processing and predictive analytics," *International Journal of Science and Research Archive*, vol. 15, no. 3, pp. 224-237, 2025. [Online]. Available:

https://www.researchgate.net/publication/392465831_Enhancing_It_incident_management_with_natural_language_processing_and_predictive_analytics

[5] Peru Study Authors, "A Machine Learning-Based Predictive Model for the Management of Incidents in Small and Medium-Sized Enterprises in Peru," in *Proc. 2024 3rd Asia Conference on Algorithms, Computing and Machine Learning (CACML '24)*, 2024. [Online]. Available:

<https://dl.acm.org/doi/10.1145/3654823.3654913>

[6] Inspirisys, "Automated Customer Ticket Classification using BERT," *Inspirisys Blog*, 2023. [Online]. Available:

<https://www.inspirisys.com/blog-details/Customer-Ticket-Classification-using-BERT/136>

[7] D. Cortiz, "Exploring Transformers in Emotion Recognition: a comparison of BERT, DistillBERT, RoBERTa, XLNet and ELECTRA," *arXiv preprint arXiv:2104.02041*, 2021. [Online]. Available: <https://arxiv.org/abs/2104.02041>

[8] F. Liu et al., "TickIt: Leveraging Large Language Models for Automated Ticket Escalation," in *Proc. 33rd ACM International Conference on the Foundations of Software Engineering (FSE '25)*, 2025. [Online]. Available: <https://arxiv.org/abs/2504.08475>

[9] S. Mittal and T. Gupta, "Comparative Analysis of Efficient Text Classification Models using Pre-Trained Transformers," *Int J Performability Eng*, vol. 21, no. 6, pp. 299-307, 2025. [Online]. Available: <https://www.ijpe-online.com/EN/10.23940/ijpe.25.06.p1.299307>

[10] Vellum AI, "Best Model for Text Classification: Gemini Pro, GPT-4 or Claude2?," *Vellum AI Blog*, 2023. [Online]. Available: <https://www.vellum.ai/blog/best-at-text-classification-gemini-pro-gpt-4-or-claude2>