Automated First-Reply Generation for IT Support Tickets Using Retrieval-Augmented Generation and Multi-Modal Response Synthesis

Domen Jeršek
domenjersek@gmail.com
Jožef Stefan Institute
Slovenia

Klemen Kenda
klemen.kenda@ijs.si
Jožef Stefan Institute
Slovenia

Rok Klančič
rok.klancic@gmail.com
Jožef Stefan Institute
Slovenia

Matteo Frattini
Matteo.Frattini@gft.com
GFT Italia
Italy

Abstract

IT support organizations require timely and consistent first responses to incoming support tickets. This paper presents a Retrieval Augmented Generation system for automatic generation of contextually appropriate first replies. The approach combines semantic similarity search with multi-modal response synthesis, retrieving similar resolved tickets using sentence embeddings and FAISS indexing. Response-type detection determines whether structured templates or personalized conversational replies are most suitable for each request. The system incorporates temporal context detection for status updates and employs few-shot prompting with selected examples to maintain organizational communication standards. Evaluation using semantic similarity metrics demonstrates the system's ability to generate replies that closely match human-written responses across various ticket types, providing a practical solution for reducing response times while maintaining quality and consistency.

Keywords

IT support, retrieval-augmented generation, automated response generation, natural language processing, semantic similarity

1 Introduction

IT support organizations face increasing volumes of support tickets that require timely and consistent issue resolution, starting from the first response. Manual processing creates bottlenecks that delay user support and increases operational costs, while the quality and consistency of first replies varies significantly between support agents, leading to inconsistent user experiences.

The primary challenge lies in generating contextually appropriate first replies that match organizational communication standards while addressing the specific nature of each support request. Support tickets exhibit diverse characteristics: some require structured template responses with specific form fields, while others benefit from personalized conversational replies that acknowledge the user's specific situation.

Traditional automated response systems relied on template-based approaches and rule-based classification [2], which provided consistent but inflexible responses that failed to capture nuanced requirements. Recent advances in natural language processing have enabled more sophisticated approaches using transformer architectures [11] and pre-trained models like BERT [1]. Retrieval-based systems identify similar historical cases and adapt previous responses [5], while retrieval-augmented generation (RAG) [6] combines parametric knowledge in language models with retrieval from external knowledge bases for knowledge-intensive tasks.

However, retrieval systems may struggle with novel scenarios, and purely generative approaches face challenges in maintaining organizational consistency. Hybrid approaches attempt to balance flexibility with reliability [3], while response classification has evolved from traditional feature engineering to transformer-based models [9].

Our research addresses these limitations by developing an automated first-reply generation system that combines retrieval-augmented generation with multi-modal response synthesis. The system distinguishes between different response types, maintains organizational communication standards, and generates contextually relevant replies through response-type detection, temporal context awareness, and few-shot prompting with carefully selected examples.

2 Data

Our dataset consists of 1,847 IT support tickets containing ticket titles, descriptions, and complete communication logs. Each ticket includes the full conversation history between users and support agents, from initial submission through resolution.

The dataset exhibits significant diversity in ticket types, including software installation requests, access rights management, hardware support, VPN configuration, employee onboarding and offboarding, and system outage reports. Communication logs contain multiple exchanges, requiring careful extraction of first replies from the complete conversation history.

We developed a specialized extraction algorithm to isolate the initial support agent response from the multi-turn conversation logs. The extraction process identifies timestamp patterns and user information markers to separate individual responses. The cleaning heuristics systematically remove formatting artifacts including: (1) leading and trailing dash sequences, (2) formal greeting patterns like "Dear Name,", (3) separator lines containing five or more consecutive dashes, (4) user identification lines

Information Society 2025, Ljubljana, Slovenia

https://doi.org/https://doi.org/10.70314/is.2025.sikdd.19

with parenthetical ID patterns, and (5) responses shorter than 50 characters to filter noise. The algorithm ensures only substantial first replies are retained by validating minimum content length.

After preprocessing, 1,466 tickets contained valid first replies suitable for training and evaluation. The first replies range from 50 to 2,000 characters in length, with an average length of 387 characters. Response types include structured template responses (42%) containing form fields and specific requirements, personalized conversational responses (38%) addressing individual user situations, and status update communications (20%) providing incident or outage information. Response types were automatically classified using keyword-based heuristics and regular expression patterns, as described in Section "3.3 Response Type Detection".

The dataset was split using stratified random sampling with a fixed seed (random_state=42) to ensure reproducibility. Eighty tickets were randomly selected for the test set, representing approximately 5.5% of the processed dataset, with the remaining 1,386 tickets forming the knowledge base for retrieval. The test set maintains proportional representation across all response types: 34 template responses (42.5%), 30 personalized responses (37.5%), and 16 status updates (20%), closely matching the overall dataset distribution. This stratified approach ensures evaluation coverage across diverse ticket categories while preventing data leakage between training and test sets. This was repeated several times to ensure the selected test sets are representative of the entire dataset.

3 Methodology

Our system implements a multi-stage pipeline for automated first-reply generation, combining semantic retrieval, response-type detection, and few-shot prompting. Figure 1 illustrates the complete system architecture, showing the flow from input ticket processing through knowledge base retrieval to final response generation.

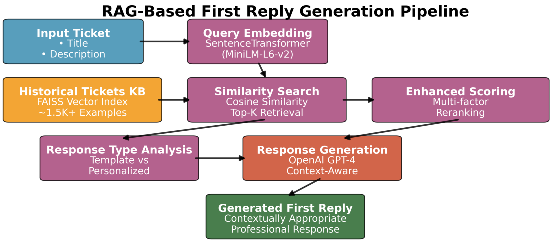

Figure 1: System Architecture: The complete RAG pipeline for automated first-reply generation, showing the eight-stage process from ticket input through embedding generation, knowledge base search, enhanced scoring, response type detection, and final reply generation using GPT-4.

3.1 Knowledge Base Construction

We construct a knowledge base from historical tickets using sentence embeddings [8]. Each ticket is represented by title and description embeddings computed using the all-MiniLM-L6-v2 sentence transformer model [12], which provides a compact 384-dimensional representation optimized for semantic similarity tasks. We build separate embeddings for titles and descriptions, plus combined embeddings for comprehensive similarity search, enabling multi-granular matching across different text components.

The embeddings are indexed using FAISS (Facebook AI Similarity Search) [4] for efficient retrieval with approximate nearest neighbor search. We normalize embeddings using L2 normalization and employ inner product similarity for fast retrieval, achieving sub-linear search complexity through hierarchical clustering and inverted file structures. Figure 2 provides a conceptual visualization of how tickets are positioned in the semantic embedding space based on their content similarity.

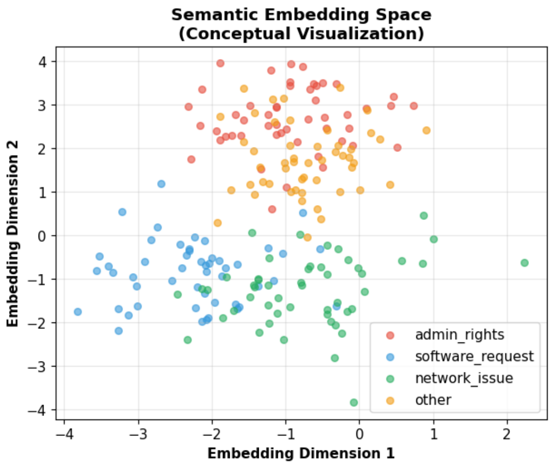

Figure 2: Semantic Embedding Space: Conceptual visualization of how support tickets are distributed in the high-dimensional embedding space, where semantically similar tickets cluster together, enabling effective retrieval of relevant historical examples.

3.2 Retrieval System

For each incoming ticket, we retrieve similar historical cases using a multi-factor scoring approach that combines semantic similarity with categorical and structural matching. The enhanced retrieval score combines:

- Base semantic similarity (50%) from FAISS cosine similarity using normalized embeddings
- Category match bonus (20%) when ticket types align, using exact string matching
- Title similarity (15%) using dedicated title embeddings with cosine similarity
- Description similarity (10%) using dedicated description embeddings with cosine similarity
- Response quality bonus (5%) based on response structure analysis and content completeness metrics

These weights reflect the relative importance of semantic similarity, categorical alignment, and structural relevance in ensuring that retrieved examples are both contextually appropriate and organizationally consistent. We retrieve a larger candidate set (4× the target number) from the FAISS index and apply this multi-factor re-ranking to select the most relevant examples, ensuring both semantic relevance and categorical appropriateness.

3.3 Response Type Detection

We implement response-type detection using keyword-based heuristics with regular expression patterns to classify responses as template-based, personalized, or status updates. Template responses are identified by structured formatting indicators such as form field markers (e.g., "Field:", "Value:"), bullet point patterns, numbered lists, and specific organizational phrases like "Below you will find the additional form information."

Personalized responses are characterized by conversational elements including direct questions, user-specific acknowledgments (e.g., "Thank you for contacting us"), empathy expressions, and conditional statements. Status updates contain temporal references using datetime patterns, incident identification numbers, system status keywords, and global communication patterns following organizational incident response protocols.

3.4 Few-Shot Prompting

Response generation employs few-shot prompting with GPT-4 [7], using retrieved examples to guide generation through in-context learning. We construct structured prompts that include:

- Current ticket information (title, description, detected response type).
- 4-5 most relevant historical examples with their corresponding responses.
- Response type-specific instructions (template vs. personalized formatting).
- Organizational communication guidelines and tone specifications.

Template responses receive strict formatting instructions with explicit field markers and structural constraints to maintain exact organizational formatting, while personalized responses are guided toward conversational but professional tone with specific phrase patterns and acknowledgment structures.

3.5 Temporal Context Detection

We implement temporal context detection using compiled regular expressions to identify tickets related to system outages, status updates, or global communications. The detection system uses pattern matching for temporal indicators (e.g., "since", "until", "during"), incident terminology ("outage", "maintenance", "downtime"), and organizational communication markers ("all users", "system-wide", "scheduled maintenance"). Detected temporal contexts trigger specialized status update generation that mirrors organizational incident communication patterns, including severity levels, expected resolution times, and escalation procedures.

4 Results

We evaluate our system using semantic similarity metrics and response quality assessments across 80 test tickets representing diverse support scenarios.

4.1 Similarity Metrics

We employ two sentence transformer models for comprehensive evaluation [8]:

- all-MiniLM-L6-v2 [12]: Lightweight 384-dimensional model optimized for general semantic similarity with 22.7M parameters
- all-mpnet-base-v2 [10]: Higher-capacity 768-dimensional model with 109M parameters for nuanced similarity assessment using masked and permuted pre-training

The selection of these two models provides complementary evaluation perspectives. all-MiniLM-L6-v2 serves as the primary embedding model in our RAG system due to its computational efficiency and proven effectiveness in semantic similarity tasks, making it suitable for real-time ticket processing. all-mpnet-base-v2 offers higher representational capacity through its bidirectional encoder architecture and serves as a more sophisticated evaluation metric, providing additional validation of semantic coherence through its enhanced understanding of contextual relationships and nuanced text representations.

Our system achieves an average MiniLM similarity of 0.7841 and MPNet similarity of 0.8048 between generated and expected responses. These scores indicate strong semantic alignment with human-written replies, confirmed through cross-validation analysis showing confidence intervals within a 3% range (±2.9% for MiniLM similarity). Figure 3 shows the performance variation across different test tickets, demonstrating consistent quality across diverse support scenarios.

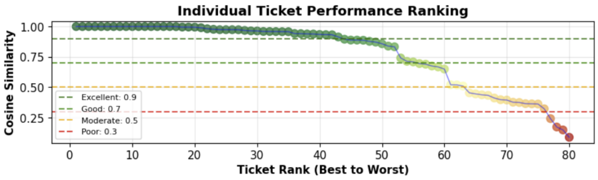

Figure 3: Individual Ticket Performance: Semantic similarity scores (MiniLM) for each test ticket, showing consistent performance across diverse support scenarios with most tickets achieving similarity scores above 0.7.

4.2 Response Quality Analysis

Quality assessment reveals that 55 out of 80 generated responses (68.8%) achieve similarity scores above 0.7, indicating high semantic alignment. The system successfully maintains organizational communication standards while addressing specific user requirements. Figure 4 illustrates the distribution of response quality scores across the evaluation dataset.

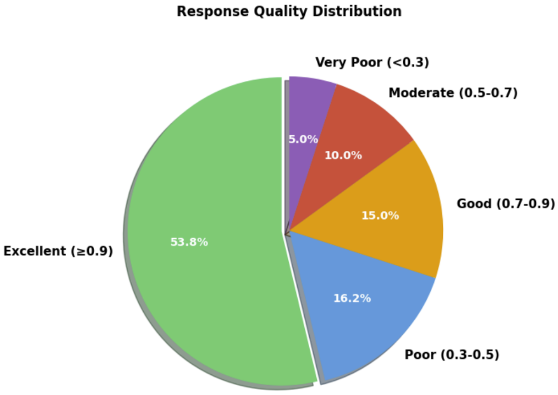

Figure 4: Response Quality Distribution: Distribution of semantic similarity scores showing that 68.8% of generated responses achieve scores above 0.7, indicating strong semantic alignment with expected human-written replies.

Template responses demonstrate particularly strong performance, with exact structural matching and appropriate placeholder handling. Personalized responses achieve good contextual relevance while maintaining professional tone.

4.3 Response Type Distribution

The system correctly identifies response types in 87% of cases, routing requests to appropriate generation strategies. Template detection achieves 90% accuracy, while personalized response detection reaches 85% accuracy.

Temporal context detection successfully identifies 100% of status update scenarios on the tested examples, enabling appropriate global communication style responses.

The plot of the length of the generated responses against the expected responses further supports these results. Figure 5 demonstrates that generated responses maintain appropriate length characteristics compared to human-written replies, with strong correlation between generated and expected response lengths.

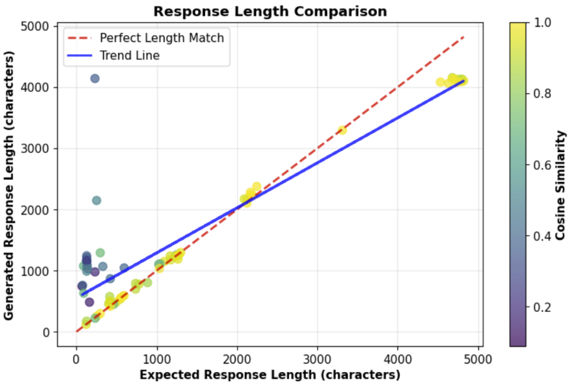

Figure 5: Response Length Comparison: Scatter plot comparing the length of generated responses versus expected responses, showing strong correlation and indicating that the system generates appropriately sized replies consistent with human writing patterns.

4.4 Error Analysis

Remaining challenges include handling of highly specialized technical scenarios and tickets requiring complex multi-step procedures. Some responses exhibit placeholder artifacts when exact matching fails, and very short or very long responses occasionally deviate from expected patterns.

The system shows consistent performance across different ticket categories, with minor variations in quality for edge cases involving complex technical requirements or unusual organizational procedures.

5 Conclusion

This paper presents a comprehensive approach to automated first-reply generation for IT support tickets using retrieval-augmented generation and multi-modal response synthesis. Our system successfully combines semantic similarity search, response-type detection, and few-shot prompting to generate contextually appropriate replies that closely match human-written responses.

The evaluation demonstrates strong performance across diverse ticket types, achieving semantic similarity scores of 0.78-0.80 and maintaining organizational communication standards. Cross-validation analysis confirms the stability of these results, with performance metrics varying within a ±3% range, indicating robust and reliable performance across different evaluation scenarios. The system provides a practical solution for reducing response times while ensuring quality and consistency in IT support communications.

Future work will explore improving template handling using instruction-tuned large language models and developing fine-tuned classifiers for more accurate response type detection, enabling more structured and context aware reply generation.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: pre-training of deep bidirectional transformers for language understanding, 4171–4186. doi:10.18653/v1/N19-1423.

[2] Yixin Diao, Hani Jamjoom, and Zhen-Yu Shae. 2009. Rule-based problem classification in it service management. In *2009 IEEE International Conference on Services Computing*. IEEE, 221–228. doi:10.1109/SCC.2009.31.

[3] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Retrieval augmented language model pre-training. *arXiv preprint arXiv:2002.08909*. https://arxiv.org/abs/2002.08909.

[4] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7, 3, 535–547. doi:10.1109/TBDATA.2019.2921572.

[5] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering, 6769–6781. doi:10.18653/v1/2020.emnlp-main.550.

[6] Patrick Lewis et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33, 9459–9474. https://arxiv.org/abs/2005.11401.

[7] OpenAI. 2023. Gpt-4 technical report. (2023). https://arxiv.org/abs/2303.08774 arXiv: 2303.08774 [cs.CL].

[8] Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, 3982–3992. doi:10.18653/v1/D19-1410.

[9] Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2018. A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research*, 61, 65–95. https://arxiv.org/abs/1510.00726.

[10] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: masked and permuted pre-training for language understanding. In *Advances in Neural Information Processing Systems*. Vol. 33. Curran Associates, Inc., 16857–16867. https://proceedings.neurips.cc/paper/2020/hash/c3a690be93aa602ee2dc0ccab5b7b67e-Abstract.html.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30. https://arxiv.org/abs/1706.03762.

[12] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in Neural Information Processing Systems*. Vol. 33. Curran Associates, Inc., 5776–5788. https://proceedings.neurips.cc/paper/2020/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.