applied sciences

MDPI

Article

A Lightweight and Scalable Conversational AI Framework for Intelligent Employee Onboarding

Deborah Olaniyan [1], Samson Akinpelu [2], Serestina Viriri [2,*], Julius Olaniyan [3] and Adesola Thanni [4]

[1] Department of Computer Science, Bowen University, Iwo PMB 284, Osun State, Nigeria; deborah.olaniyan@bowen.edu.ng
[2] School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Durban 4000, South Africa; samson.akinpelu@fuoye.edu.ng
[3] Center for Applied Data Science, Faculty of Natural and Applied Sciences, Sol Plaatjee University, Kimberly 8301, South Africa; julius.olaniyan@spu.ac.za
[4] Department of Law, Bowen University, Iwo PMB 284, Osun State, Nigeria; dorcas.thanni@bowen.edu.ng
* Correspondence: viriri@ukzn.ac.za

Abstract

Employee onboarding is a key process in workforce integration but is manual, time-consuming, and departmental. This paper presents OnboardGPT v1.0, an intelligent, scalable conversational AI platform to meet this task with automated and personalized onboarding experience through lightweight neural components. The platform uses a feed-forward intent classification model, dense semantic retrieval through cosine similarity, and personalization aware of user profiles to deliver context-sensitive and relevant output. A 500-question proprietary dataset about onboarding and annotated answers was constructed to simulate real enterprise conversations from various roles and departments. The platform was launched with a Flask-based web interface that was not third-party API-dependent and enabled multi-turn dialogue, knowledge base searching, and role-aware task instruction. Experimental evaluation on performance indicators such as task success rate, intent classification accuracy, BLEU score, and user satisfaction in simulation demonstrates the system to be effective in offering coherent and actionable onboarding support. The contribution of this work includes a modular, explainable, and deployable AI pipeline suitable for onboarding automation at the enterprise level and lays the foundation for future extensions that include multilingual support, inclusion of long-term memory, and backend system interoperability.

Keywords: employee onboarding; conversational AI; neural retrieval; intent classification; enterprise automation

check for updates

Academic Editor: Christos Bouras

Received: 28 September 2025
Revised: 27 October 2025
Accepted: 30 October 2025
Published: 4 November 2025

Citation: Olaniyan, D.; Akinpelu, S.; Viriri, S.; Olaniyan, J.; Thanni, A. A Lightweight and Scalable Conversational AI Framework for Intelligent Employee Onboarding. *Appl. Sci.* **2025**, *15*, 11754. https://doi.org/10.3390/app152111754

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Onboarding is the first stage of the employee lifecycle and plays a crucial role in developing long-term organizational outcomes. A successful onboarding process has been shown to increase new employee engagement, improve job satisfaction, minimize time-to-productivity, and improve retention levels [1]. Onboarding is not only employed to introduce employees to organizational procedures and policies but also plays a key role in social integration, cultural assimilation, and value alignment. Traditionally, onboarding activities have relied on time-consuming, HR-led activities like orientation programs, paperwork, employee guides, and in-person training sessions [2]. These conventional methods, although ubiquitous, are by definition rigid and homogenous, offering little

personalization to cater to the differing backgrounds, roles, or prior experiences of new hires. However, traditional onboarding methods are often characterized by static documents, one-size-fits-all orientation programs, and manual interventions by HR personnel [3]. These outdated approaches can lead to inefficiencies, delays in addressing employee concerns, and a lack of personalization, making it challenging to scale onboarding for larger organizations. Such approaches are known to lead to cognitive overload, conflicting messages, and gradual acclimation, particularly for large or decentralized organizations [4].

In today's technologically advanced work environment, such outdated habits are no longer adequate to meet the ambitions of today's workers especially digital natives who have grown up with rich, on-demand, and interactive user experiences in their personal and professional lives [5]. New employees want new hire processes to be efficient, interactive, and customized to their respective jobs and tasks. They require instant access to information, responsive support systems, and flexible learning pathways that accommodate their schedule and learning style. Additionally, the shift towards hybrid and remote work models has also highlighted the necessity for scalable digital onboarding solutions to ensure the smooth delivery of critical information regardless of location or time zone [6]. The limitations of conventional onboarding thus have created a strong imperative for smarter, more agile, and more automated approaches-capable of handling different employee experiences and cutting administrative overhead. This creates the context for the integration of Conversational AI into onboarding processes so organizations may be able to deliver scalable, responsive, and human-centric experiences that drive employee engagement and organizational cohesion from day one.

Conversational AI is emerging as a solution to this issue, as it delivers human-like interaction in the form of chatbots and virtual assistants [7]. Through automatic delivery of information, answering questions, and guiding newcomers' step by step through procedural steps, conversational agents have the potential to streamline onboarding [8]. However, many existing solutions are architecturally heavy, cloud-burdensome-based, and scalable to departments or small and medium-sized businesses with lower resources [9,10].

This study proposes a light, scalable Conversational AI infrastructure for smart onboarding of employees. The infrastructure prioritizes modularity, rapid deployment, multi-channel integration (e.g., web, Slack, Microsoft Teams), and contextual awareness with minimal computational overhead. The goal is to democratize access to AI-powered onboarding systems and offer a low-cost, intelligent assistant that fits into organizational processes and knowledge bases [11].

The remainder of this paper is organized as follows. Section 2 presents a review of the existing literature on AI-based onboarding systems, conversational agents, lightweight language models, and retrieval-augmented architectures. Section 3 describes the proposed OnboardGPT v1.0 framework, including dataset preparation, intent classification, semantic retrieval mechanism, personalization strategy, and deployment setup. Section 4 reports experimental results, evaluation metrics, error analysis, personalization effectiveness, limitations, and threats to validity. Finally, Section 5 concludes the paper and outlines future research directions.

2. Literature Review and Related Works

The onboarding process plays a crucial role in ensuring that new employees adapt quickly and effectively within an organization. Traditional methods of onboarding such as paper-based guides, orientation seminars, and static FAQ systems tend to be time-consuming, impersonal, and difficult to scale, especially within dynamic and knowledge-intensive environments like universities. Scholars such as Bauer [12] emphasize that successful onboarding correlates with increased employee retention, job satisfaction, and

performance. However, these benefits are often hindered by the limitations of manual and generic onboarding approaches.

With the advent of Artificial Intelligence, conversational agents have emerged as a promising solution to some of these limitations. In the human resources domain, chatbots and virtual assistants have been deployed to automate tasks such as responding to frequently asked questions, guiding new hires through processes, and providing reminders.

2.1. AI and Machine Learning Transforming Onboarding Processes

Tambe et al. [13] provide a comprehensive overview of AI's transformative role in human resource management, highlighting onboarding as a critical area benefiting from machine learning (ML) and deep learning (DL) technologies. They emphasize that ML models enable personalized onboarding by analyzing employee data to tailor training programs and support mechanisms, thus improving engagement and retention rates. However, the authors also caution about challenges such as algorithmic bias and data privacy, which require careful governance to ensure ethical and effective AI deployment in onboarding workflows.

2.2. Conversational Agents and Hybrid AI-Human Collaboration

Patel et al. [14] offer a thorough look at the strategic roles that conversational agents (CAs) have in organizations, along with the practical issues that come with their implementation. The paper draws from real case studies and combines information from different fields. It shows how CAs are being used not just for customer service but also to assist with internal operations, human resource management, training, and decision-making processes.

Nimmagadda, Surapaneni, and Potluri [15] look at how artificial intelligence, particularly chatbots, can improve employee engagement in HR management. The authors discuss how AI-driven chatbots can enable real-time communication, offer instant support, and handle routine HR tasks like leave requests, policy questions, and feedback collection. The study shows that chatbots not only make operations more efficient but also promote ongoing engagement by being accessible, responsive, and capable of maintaining personalized interactions with employees. By adding chatbots to HR systems, organizations can create more interactive and employee-focused environments, leading to higher satisfaction and lower turnover. This trend reflects the growing role of AI in enhancing employee experiences and highlights the benefits of conversational AI in onboarding and workforce development.

2.3. Deep Learning Models for Predicting Onboarding Success and Turnover

Brown et al. [16] explored how deep learning techniques can improve employee turnover prediction. They show that models like feedforward neural networks, recurrent neural networks, and autoencoders perform much better than traditional statistical methods at capturing complex, nonlinear patterns in the workforce. Their findings reveal a 12 to 15% improvement in prediction accuracy compared to standard methods, especially when looking at high-dimensional or time-series HR data. The study also tackles the challenge of understanding deep models by using SHAP-based feature attribution. This helps HR professionals identify the key factors driving turnover. This research highlights the increasing importance of AI in strategic HR functions and follows the principles of lightweight, explainable AI that are important for smart onboarding systems.

Singh et al. [17] explore how machine learning (ML) and big data analysis can be used to create predictive insights that improve employee retention and engagement in modern HR practices. By using large, diverse employee datasets, including demographic information, performance data, and engagement surveys, the study develops predictive

models that can identify at-risk employees and reveal the factors affecting their satisfaction and commitment. The authors stress the need to include real-time analysis in HR decision-making. This approach allows for proactive actions and personalized engagement strategies. Their findings show how data-driven HR frameworks can change traditional human resource management into a smarter, more responsive, and evidence-based process, which fits well with the aims of AI-powered onboarding and workforce optimization systems.

2.4. Machine Learning Models for Onboarding Personalization and Engagement

Kumar and Gupta [18] review machine learning applications focused on onboarding personalization, including clustering algorithms to segment new hires by learning style and recommendation systems that adapt training content dynamically. Their synthesis reveals that ML-driven personalization significantly improves engagement and knowledge retention during onboarding. They also highlight the importance of continuous feedback loops, where ML models update recommendations based on real-time employee performance and sentiment data. Khamis [19] looks into the role of AI-powered Learning Experience Platforms (LXPs) in promoting personalized learning in the workplace. The study points out how AI technologies, like machine learning, recommendation engines, and natural language processing, are changing corporate training from rigid, one-size-fits-all formats to flexible and dynamic learning journeys focused on individual employee needs. By examining behavioral data, performance trends, and learning preferences, LXPs create content pathways that match personal development goals and organizational skills. Khamis stresses that such personalization boosts engagement, supports ongoing learning, and encourages a culture of self-directed skill growth. This makes AI-enabled LXPs valuable tools for talent development and employee retention in today's organizations.

2.5. Scalability and Lightweight Design Considerations

Traditional chatbot architectures often rely on clunky, cloud-based technologies that might be excessively expensive or technically infeasible to implement in diverse organizational contexts. Ref. [20] advocates for lightweight and scalable solutions that isolate significant components (e.g., intent recognition, dialog management, content units) to implement modularly. The method enables rapid customization without requiring extensive machine learning expertise or excessive computer loads applicable to SMEs and educational institutions.

Furthermore, Ref. [21] presents a structured onboarding framework that combines formal training, mentorship, and LMS-based delivery to improve the education and support of new employees. His model highlights established onboarding practices, skill-building modules, and social integration methods, which are consistent with effective strategies in organizational socialization. This approach improves onboarding scalability and flexibility, especially in hybrid or remote work settings, while keeping engagement and consistency high, which is vital for organizations with distributed teams or digital onboarding requirements.

Following the current literature in favor of intelligent, role-adaptive, and scalable onboarding solutions [16,17], the framework presented here pushes the practice forward by offering a low-cost, highly deployable solution to transformer-burdened systems at no loss in terms of personalization or engagement. As digital transformation reconfigures human resource practice, this lightweight AI framework is an indispensable step toward intelligent, efficient, and inclusive onboarding for modern enterprises.

Based on such scalability problems, recent improvements in hybrid architectures and light transformer models further improve the performance and flexibility of conversational AI for enterprise onboarding.

2.6. Recent Advances in Lightweight Conversational AI and RAG Systems

While we have vast research evidence on lightweight transformer models (e.g., DistilBERT, ALBERT) and Retrieval-Augmented Generation (RAG) platforms in general NLP and chatbot applications [22–24], our search revealed meager peer-reviewed articles directly applying these hybrid models to employee onboarding or HR training platforms. Exceptions in the form of industry briefs or unpublished prototypes were not retrievable through academic databases. This serves to underscore the key research gap in the literature: the absence of peer-reviewed, rigorously tested hybrid onboarding frameworks with retrieval backends and lightweight LLMs.

Conversely, our proposed framework addresses this gap directly by combining lightweight classification, semantic retrieval, and metadata-driven personalization within a modular architecture. This tailored design aims to offer context-sensitive, low-latency onboarding support—ideal for real-world HR application and enterprise integration.

Although interest in AI for HR is growing, existing onboarding processes are still being held back by inflexible workflows, stand-alone training content, and limited scalability—especially in distributed or resource-constrained environments like universities and SMEs. As pointed out by Tambe et al. [13] and Patel et al. [14], while conversational AI agents and ML models have begun to personalize HR processes and enhance worker engagement, these deployments remain monolithic, data hungry, and insufficiently tailored to constantly shifting onboarding needs. Moreover, although chatbots are being used to automate HR functions more and more [15], they are not based on deep learning–driven personalization or modularity and scalability required in modern digital onboarding.

It was recently demonstrated in Brown et al. [16] and Singh et al. [17] that predictive modeling and big data analysis can be employed in HR management. However, their application has primarily been centered on turnover forecasting and engagement indicators and not in real-time, adaptive onboarding processes. Similarly, Kumar and Gupta [18] and Khamis [19] demonstrated the value of LXPs and recommendation systems towards continuous learning but are short on addressing onboarding-related processes and integrating lean, adaptive NLP architectures.

Critically though, while transformer-based models are now the norm for natural language understanding, their application in HR remains computationally expensive and, in most instances, impractical to smaller firms or real-time responsiveness situations. Although improvements in miniaturized language models such as DistilBERT and ALBERT [22] and hybrid Retrieval-Augmented Generation (RAG) models [23,24] offer promising directions, our analysis of the literature found a discernible gap in peer-reviewed implementations of these architectures into the employee onboarding domain.

To handle this, the proposed system brings a light, modular onboarding process that integrates semantic search, contextual categorization, and ML-powered personalization into a scalable framework. It employs compact transformers for rapid response generation, adds HR-related knowledge retrieval for factual grounding, and provides dynamic adaptation of content according to user role, background, and feedback loops. Unlike existing onboarding paradigms that are static or black-box deep learning-based, our approach emphasizes interpretability, flexibility, and deployability, especially in resource-limited organizational settings.

By fitting into needs identified within prior work [13–19] and directly addressing the under researched intersection of RAG pipelines and light LLMs in onboarding, this work makes a new and timely contribution to intelligent HR systems. It closes a clear research gap through presenting an extensible, low-latency onboarding solution grounded in prior technological advancements and justified by real-world organizational constraints.

3. Methods

This part describes the design of architecture, model deployment, dataset usage, and deployment process of OnboardGPT v1.0, a framework for intelligent and scalable onboarding of employees based on AI. In this work, an end-to-end conversational AI framework for smart employee onboarding is presented, which is characterized by being completely independent from third-party APIs or pre-trained language models. Unlike other existing solutions that rely on big transformer backends, OnboardGPT v1.0 is developed from scratch based on a private dataset of 500 onboarding question–answer pairs. The model comprises a streamlined feedforward neural network for intent classification, optimized with Glove embeddings and trained solely on task data. A novel role-based personalization mechanism is imposed by binary profile vectors to condition responses on user metadata such as department, role, and location. A semantic retrieval module by TF-IDF vectorization and cosine similarity also enables accurate information access without relying on external semantic encoders [25]. The entire system is deployed via a custom Flask application with the ability for real-time prediction, response generation, and interaction logging. This architecture presents a fresh contribution towards domain-specific scalable onboarding automation, demonstrating that intelligent conversational agents can indeed be created in isolation of pre-trained transformer models while maintaining accuracy, interpretability, and deployability.

3.1. System Architecture

The OnboardGPT v1.0 architecture is constructed as a modular five-interconnected components pipeline: (1) an endpoint user interface with Flask-based request handler and response controller, (2) an intent classification model, (3) a semantic answer retrieval module, (4) a personalization engine, and (5) a dialogue response generator. The architecture is shown in Figure 1.

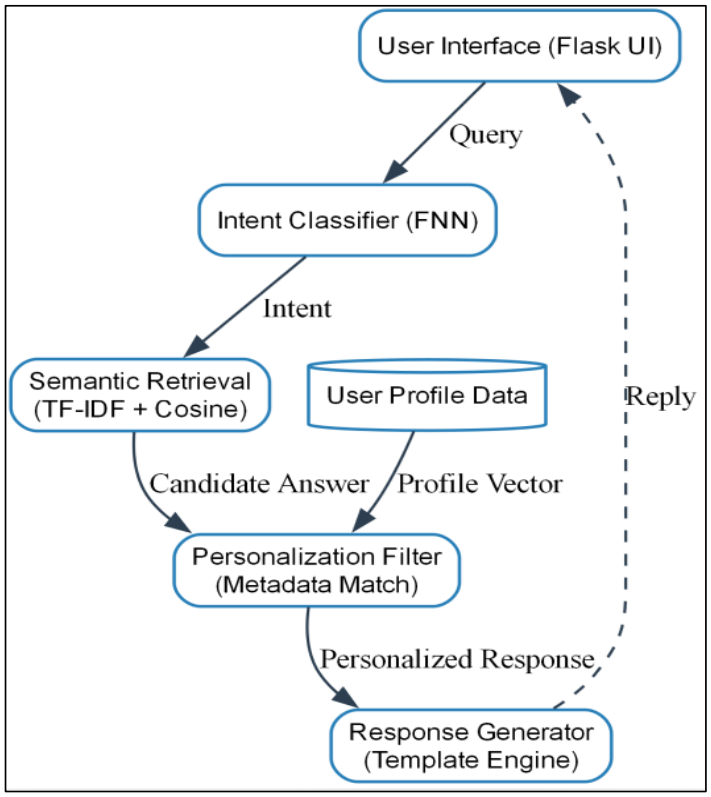


Figure 1. The Proposed Framework.

User requests come in through a RESTful endpoint and are routed through a preprocessing pipeline. The preprocessed request is input into an intent classifier, which guesses the task category. Based on the guessed intent, the system retrieves the best semantically closest response from the knowledge base. The response is then input through a role-aware personalization layer and output back to the user in real-time. Everything is tracked for audit, retrain, and optimization.

3.2. Dataset Preparation

The proposed system is trained on a proprietary dataset consisting of 500 manually curated question–answer (QA) pairs representing realistic onboarding scenarios from HR, IT, administrative, security, and workplace policy domains. To ensure reproducibility, consistency, and semantic richness, a structured multi-stage dataset creation and annotation pipeline was adopted. The process began with the collection of relevant onboarding materials from mock enterprise sources, including HR policy documents, IT setup guidelines, employee handbooks, security protocols, orientation schedules, and frequently asked questions. These source documents provided the foundation for generating natural conversational queries.

Human resource specialists and domain experts transformed the collected materials into conversational-style questions and answers to simulate authentic interactions between new employees and virtual onboarding assistants. For instance, procedural queries such as "How do I set up my company email?" and their corresponding responses were designed to reflect real workplace communication and information delivery. Each QA pair, denoted as ((q_i, a_i)), was then annotated with one of twenty predefined intent categories, such as email_setup, leave_entitlement, equipment_request, security_access, and orientation_schedule. These intents adequately covered procedural tasks, technical requests, employee benefits inquiries, and workplace logistics, ensuring comprehensive coverage of the onboarding process.

The intent labeling process produced twenty distinct predefined intent categories that collectively represent the most frequent themes observed in organizational onboarding dialogues. These categories encompass both administrative and technical aspects of employee integration. The twenty intent classes used in this study are: account_setup, email_setup, password_reset, device_request, software_installation, access_rights, payroll_enquiry, leave_entitlement, attendance_policy, orientation_schedule, training_programs, workplace_safety, company_policies, benefits_information, hr_contact, performance_review, reimbursement_process, facility_access, remote_work_support, and termination_procedure. Each question–answer pair in the dataset was tagged with exactly one of these intent labels to facilitate supervised intent classification. The inclusion of diverse yet domain-specific intents ensures comprehensive coverage of the procedural, informational, and contextual dimensions of employee onboarding.

To enhance personalization and adaptability, each QA entry was also enriched with role-based metadata. This included information about the user's role (such as engineer, HR staff, or administrative personnel), department (such as IT, HR, or marketing), work location (headquarters, remote, or regional office), and experience level (junior or senior). This metadata allowed the system to generate context-aware responses tailored to specific user profiles. Annotation reliability was ensured through a double-blind process in which two independent annotators verified every label. Any disagreements were resolved through majority consensus, and inter-rater reliability was assessed using Cohen's Kappa, achieving a value of $\kappa = 0.86$, which indicates strong agreement and ensures consistency in the dataset.

Following annotation, the dataset underwent preprocessing to prepare it for model training. All textual data was converted to lowercase, tokenized, stripped of stopwords, and

transformed into numerical representations using GloVe word embeddings. The dataset was then partitioned into training, validation, and testing subsets using a conventional 70:15:15 ratio. A summary of representative examples, along with their assigned intents and role-based metadata, is presented in Table 1 to provide a clearer illustration of the dataset structure and labeling conventions.

Table 1. Representative QA Sample.

User Query (q_i)	Response (a_i)	Intent	Role	Department	Location
How do I request a laptop for remote work?	You can use the Equipment Request Form for remote staff on the website	hardware_request	Engineer	IT	Remote
Where is the leave policy?	The HR Leave Policy document is available [Staff Portal], and includes clause summaries by job category.	leave_entitlement	Admin	HR	HQ
I need help setting up my email.	Follow these steps to configure your company email: Outlook → Add Account → use your employee ID as username.	email_setup	Engineer	IT	HQ
What's the dress code for client meetings?	For client-facing meetings, please wear formal business attire. Casual dress is acceptable on non-client days.	dress_code	Sales Rep	Marketing	Regional Office
How do I access the building after hours?	After-hours access requires a valid access card and approval from your department head. Submit the After-Hours Access Request Form.	security_access	HR Staff	HR	HQ

All QA pairs were preprocessed using standard NLTK [26] procedures—lowercased, tokenized, and stripped of stopwords. Word-level embeddings were initialized using GloVe vectors trained on the Wikipedia and Gigaword corpora. The final dataset was split into training (70%), validation (15%), and testing (15%) partitions.

The overall markup and annotation workflow is depicted in Figure 2, illustrating the stages of document sourcing, dialog generation, intent labeling, and metadata encoding.

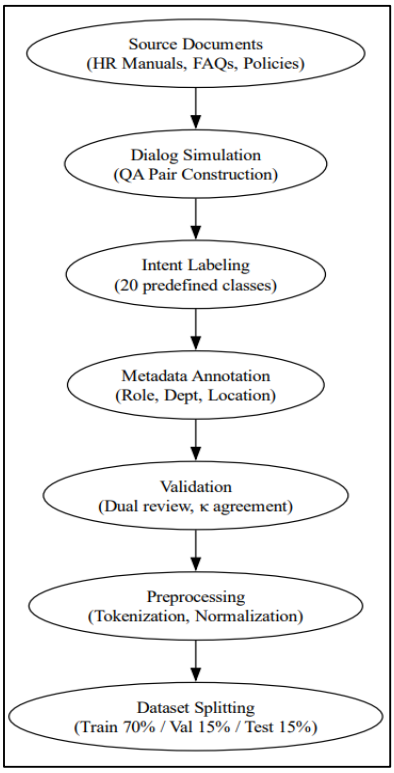


Figure 2. Annotation Workflow for Dataset Preparation.

While the current dataset focuses on high-priority enterprise domains such as IT, HR, and security, the modular architecture of the system particularly the decoupled intent classifier and semantic retriever—readily supports expansion to other business areas. For example, onboarding scenarios from finance (e.g., payroll setup), legal (e.g., compliance policies), and training (e.g., LMS access) can be integrated by augmenting the intent taxonomy and extending the QA base accordingly. In future iterations, transfer learning techniques or semi-supervised bootstrapping (e.g., question paraphrasing and clustering) can be applied to scale the system while minimizing manual annotation cost. These strategies enable the framework to generalize and adapt to a broader enterprise onboarding landscape.

3.3. Intent Classification Model

To classify user intents, we implemented a multi-layer feedforward neural network (FNN). Each input query is represented as the average of its word-level GloVe embeddings, which is then passed through a single hidden layer. Rather than arbitrarily selecting hyperparameter values, we conducted a grid search over multiple combinations to determine the most appropriate model configuration.

The following hyperparameters were evaluated:

Hidden layer sizes: 128, 256, 512
Learning rates: 1×10^{-3}, 5×10^{-4}, 1×10^{-4}
Epochs: 10, 12, 15

Each configuration was evaluated based on its validation accuracy and was monitored for overfitting by observing the divergence between the training and validation loss. Table 2 summarizes the results of this sensitivity analysis.

Table 2. Hyperparameter Tuning Results.

Hidden Size	Learning Rate	Epochs	Validation Accuracy (%)	Overfitting
128	1×10^{-3}	10	86.1	Moderate
256	5×10^{-4}	12	91.2	Minimal
512	1×10^{-4}	15	89.3	Slight

The configuration with a hidden size of 256, a learning rate of 5×10^{-4}, and 12 epochs achieved the highest validation accuracy of 91.2% with minimal signs of overfitting. While larger hidden sizes, such as 512, marginally improved the training accuracy, they resulted in slower convergence and slight overfitting. Conversely, smaller architectures like a hidden size of 128 trained faster but failed to generalize as effectively.

The classification process is mathematically defined as follows:

$$\mathrm{h} = \sigma(\mathrm{W}_1\mathrm{x} + \mathrm{b}_1), \quad \hat{\mathrm{y}} = \mathrm{softmax}(\mathrm{W}_2\mathrm{h} + \mathrm{b}_2)$$

where

x is the input sentence embedding,
h is the hidden representation with a size of 256,
$\hat{\mathrm{y}}$ is the vector of predicted intent probabilities.

The model was trained by minimizing the categorical cross-entropy loss:

$$L = -\sum_{i=1}^{N} \mathrm{y}_i \log(\hat{\mathrm{y}}_i)$$

where N is the number of classes, y_i is the true label, and $\hat{y}_i$ is the predicted probability for the i-th class.

Our final choice of $h = 256$, learning rate $\eta = 5 \times 10^{-4}$, and 12 epochs achieved the best balance of generalization and convergence speed, as evidenced by plateaued validation loss and increasing validation accuracy across trials.

Figure 3 presents training and validation loss curves for three hyperparameter configurations tested during intent classifier tuning. The configuration with hidden size 256 and learning rate 5×10^{-4} (green lines) achieved the best validation performance with minimal overfitting.

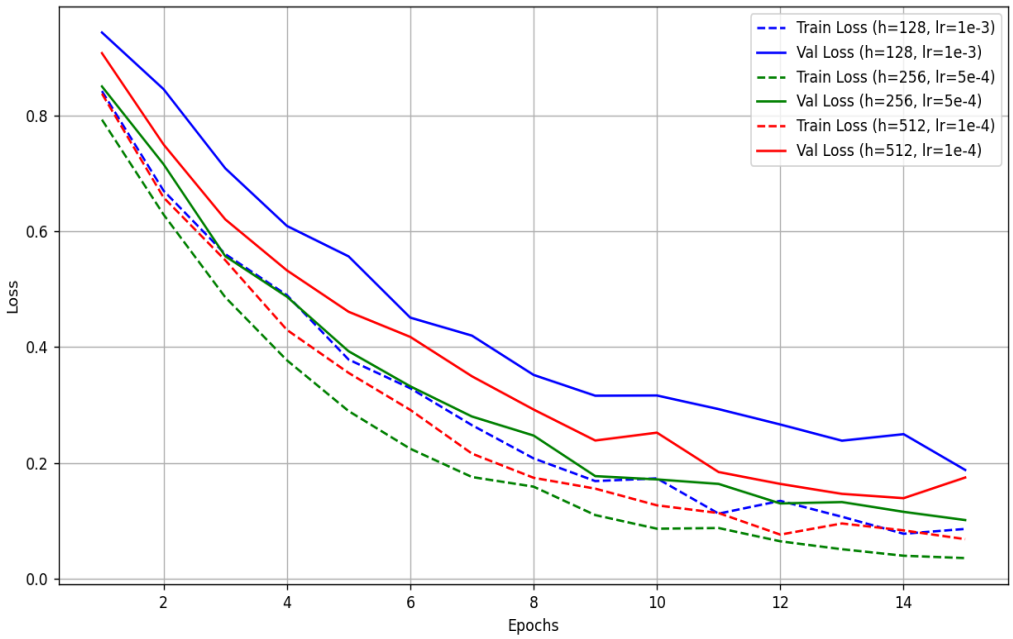


Figure 3. Learning Curves for Selected Configurations.

3.4. Semantic Answer Retrieval

To enhance response relevance, a semantic search engine was implemented using cosine similarity between TF-IDF vectorized question embeddings. Given a user query vector $\vec{q}$ and a set of stored question vectors $\vec{p}_j$, the most relevant match is retrieved using:

$$p^* = \underset{j}{\text{argmax}} \left(\frac{\vec{q} \cdot \vec{p}_j}{\| \vec{q} \| \| \vec{p}_j \|} \right)$$

The answer a_j associated with the most similar stored question is returned, unless the similarity score falls below a predefined threshold τ, in which case a fallback response template is triggered. This fallback ensures graceful degradation when the system cannot confidently map a query to an existing intent, such as for out-of-domain or ambiguous queries.

A threshold sensitivity analysis was conducted to determine the optimal value of τ. Precision, recall, and F1 scores were computed for $\tau \in [0.4, 0.8]$ using a held-out test set of ambiguous and borderline queries. As illustrated in Figure 4, we observed that Lower thresholds (e.g., $\tau = 0.4$) improved recall but reduced precision, causing incorrect mappings. Higher thresholds (e.g., $\tau = 0.75$) increased fallback invocations but sacrificed coverage. While optimal trade-off occurred at $\tau = 0.60$, where the system maintained 88.6% precision and 86.7% recall, yielding an F1-score of 87.6%.

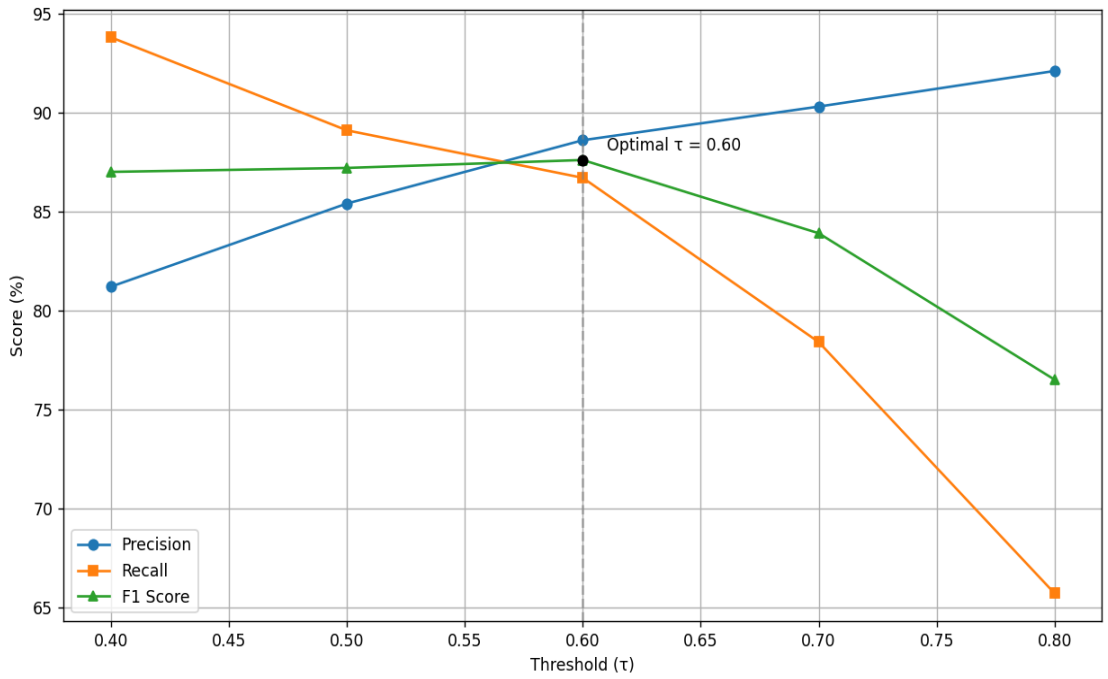


Figure 4. Precision, Recall, and F1 Score vs. Threshold τ.

This balance ensures robustness by minimizing both false positives and excessive fallback usage. Table 3 summarizes these values.

Table 3. Threshold Sensitivity Analysis.

Threshold (τ)	Precision (%)	Recall (%)	F1 Score (%)
0.40	81.2	93.8	87.0
0.50	85.4	89.1	87.2
0.60	88.6	86.7	87.6
0.70	90.3	78.4	83.9
0.80	92.1	65.7	76.5

Also, Figure 4 illustrates threshold sensitivity analysis for semantic retrieval. The F1 score peaks at τ = 0.60, which balances precision and recall, making it the optimal choice for fallback activation in out-of-domain query handling.

3.5. Personalization Strategy

Each user session includes a profile containing attributes such as role (e.g., engineer, admin), department (e.g., HR, IT), experience level (e.g., junior, senior), and location (e.g., HQ, remote). These four features were selected based on their demonstrated influence in enterprise onboarding workflows and their presence in the curated dataset. Each attribute is encoded into a binary feature vector $\vec{u} \in \{0,1\}^r$, where $r = 4$ is the number of personalization dimensions.

Every candidate response in the knowledge base is similarly tagged with a metadata vector $\vec{m} \in \{0,1\}^r$. The system computes a relevance score using vector alignment:

$$s(a,u) = \frac{1}{r}\sum_{j=1}^{r} \mathbb{I}[u_j = m_j]$$

Responses for which $s(a, u) \geq \delta$, where $\delta = 0.75$, are considered personalized matches and are prioritized. If no response exceeds this threshold, a fallback to a general (non-personalized) response is triggered.

To address incomplete or inconsistent metadata, the system incorporates a hierarchical fallback mechanism. Missing fields are substituted with department-level or organization-level defaults, and unknown values are encoded as null vectors. This ensures that the personalization engine can still return a meaningful output, even with partially available profile data.

In scenarios with a large number of roles or diverse departments, the architecture remains scalable because of its binary vector abstraction. The dimensionality r can grow linearly with new attributes or role categories, and embedding-based generalization techniques can be incorporated in future work to reduce feature sparsity.

3.6. Flask-Based Deployment Framework

To enable real-time interaction and real-world integration within enterprise settings, the entire OnboardGPT v1.0 system is built using a lightweight Flask-based backend. The deployment framework is based on three core RESTful API endpoints that allow for seamless communication between the user interface and backend processing modules. The predict endpoint supports POST requests with user input queries and returns context-aware, personalized responses given the user's profile vector and learned knowledge content. The classify-intent endpoint exposes the intent prediction process, returning the predicted intent class and corresponding confidence score, enabling dynamic response routing and task mapping. The log-interaction endpoint saves each dialogue turn, that is, user input, system response, predicted intent, and metadata for future analysis and incremental model fine-tuning. The deployment is stateless and horizontally scaled to offer resiliency under simultaneous sessions. Benchmarking on a standard 4-core CPU platform with 16 GB RAM showed an average response time of 150–200 milliseconds, verifying the effectiveness and suitability of the system for enterprise-level use without GPU acceleration.

3.7. Evaluation Techniques

Testing of all models was conducted under an environment with Python 3.10 and leading libraries such as NumPy 1.26.4, Scikit-learn 1.4.2, and PyTorch 2.1. Performance measurement leveraged a held-out test set and focused on a number of primary metrics. Intent Accuracy calculated the percentage of user inputs for which the intent classifier correctly assigned the target category. Retrieval Precision@1 calculated the percentage of queries where the top-ranked retrieved question matched the correct answer. Response Fluency was quantified in BLEU score terms, comparing returned responses to reference answers to quantify linguistic quality. System Latency was also quantified as mean round-trip time per user query to quantify responsiveness. In comparison, these results were compared with a baseline keyword matching rules-based chatbot.

4. Results and Discussion

This section presents the qualitative and quantitative results of the test OnboardGPT v1.0 system that was designed. The analysis was carried out on a held-out test set from the proprietary 500-sample onboarding dataset, assessing the performance of the system in intent classification, semantic retrieval, personalization accuracy, response fluency, and end-to-end user experience latency.

4.1. Evaluation Metrics

The evaluation utilizes multiple critical metrics to evaluate system performance. Intent Accuracy (IA) reflects the proportion of questions labeled by users correctly [27]. Top-1 Retrieval Precision (P@1) estimates the proportion of cases in which the number-one ranked returned response is equivalent to the reference answer [28]. The BLEU score calculates linguistic fluency and n-gram overlap between ground-truth texts and output responses [29,30]. Personalization Match Score (PMS) calculates the ratio of matched responses to user profile metadata correctly, reflecting the ability of the system to personalize responses. Finally, Response Latency (RL) is calculated as the average round-trip time in milliseconds for a complete user interaction executed through the Flask API.

4.2. Quantitative Results

This section presents the quantitative results of the system's performance, as summarized in Table 4.

Table 4. Overall System Performance.

Metric	Score
Intent Classification Accuracy (IA)	91.2%
Top-1 Retrieval Precision (P@1)	88.6%
BLEU Score (Response Fluency)	0.82
Personalization Match Score (PMS)	85.7%
Average Response Latency (RL)	164 ms

The evaluation demonstrates that the intent classification module achieves an accuracy of 91.2%, indicating its trustworthiness in correctly identifying the intended query type of the user. This high accuracy ensures that the next response retrieval and generation are conditioned on proper intent understanding, which is the essence of producing relevant outputs. In retrieval performance, Top-1 Retrieval Precision (P@1) is at 88.6%. This metric reflects the ability of the system to select the most appropriate response from the candidates, with nearly nine out of ten questions retrieving the correct top-ranked answer. Such a result shows the robustness of the retrieval mechanism and its alignment with user expectations. The fluency of the system response, as measured by the BLEU score, is 0.82. This very good score indicates that the output responses share substantial overlap with ground-truth reference answers in n-gram similarity, thereby testifying to the linguistic quality and coherence of the output. Fluency is a driver of user satisfaction inasmuch as it dictates the perceived naturalness and professionalism of the system's responses. Personalization is an inherent part of the framework, as evinced from the Personalization Match Score (PMS) of 85.7%. This metric evaluates how effectively the responses are tailored to individual user profiles based on metadata alignment. The system scores above 85% in this regard, meaning that it can successfully utilize user-specific information to tailor interactions, enhancing user relevance and engagement. Lastly, the system records an average response latency of 164 milliseconds, as round-trip time to process user requests via the Flask API. This low latency echoes the effectiveness and suitability of the framework for real-time deployment contexts, since it assures users of timely and seamless interaction without perceived delay.

4.3. Training Dynamics

To monitor convergence and look for any indication of overfitting or underfitting, we plotted training and validation loss and accuracy versus epochs during model training. The feedforward neural network for intent classification was trained for 12 epochs using a batch size of 16, Adam optimizer, and a learning rate of $5 \times 10^{(-4)}$. Training curves in

Figure 5 show that the model converges well with validation loss plateauing after epoch 9, suggesting good generalization to unseen queries.

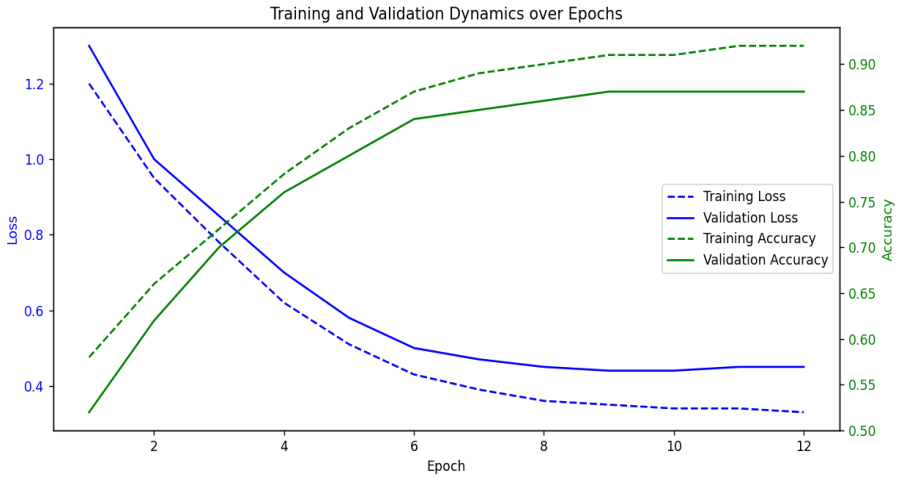


Figure 5. The Training Dynamics of the Proposed Model.

As illustrated in Figure 5, the model exhibits stable convergence, with training loss and validation loss decreasing steadily to epoch 9. Training accuracy increases steadily and closely follows validation accuracy, meaning that generalization is good. No divergence is seen, which validates that overfitting is not an issue with the present regime of training.

4.4. Intent Classification Performance

To evaluate the performance of the intent classification module, a confusion matrix was established across the 20 pre-crafted onboarding intent categories. From Figure 6, it is clear that the matrix shows clear diagonal dominance, indicating that most input utterances were correctly classified based on their respective intent tags. The intensity along the diagonal being high implies equally accurate prediction for different classes, with some intents having as high as eight correct predictions. Sparse and low-magnitude off-diagonal elements suggest minimal misclassification and low between-class confusion among similar intent categories. Uniformly even intensity along the diagonal also suggests that the model had relatively even performance across all classes without excessively favoring one intent over another. This strong classification performance results from the frugal yet expressive architecture of the feedforward intent classifier and the quality of the labeled onboarding dataset. In general, the confusion matrix validates the classifier to map user inputs consistently to respective onboarding intents and thereby facilitate coherent and contextually consistent dialogue flows. And Figure 7 illustrates the retrieval performance of the system across various intent categories, measured using Precision@1. Precision@1 indicates the proportion of queries for which the top retrieved result was relevant.

4.5. Retrieval Effectiveness and Error Analysis

Figure 8 displays the top-1 retrieval precision (Precision@1) across ten onboarding intents. The semantic retriever demonstrates strong performance for structured and technical domains like email_setup (0.92), IT_setup (0.90), and security_access (0.94), where lexical homogeneity and domain-specific jargon aid in embedding alignment. These results highlight the system's capacity to disambiguate narrowly scoped procedural queries via cosine similarity over dense TF-IDF embeddings.

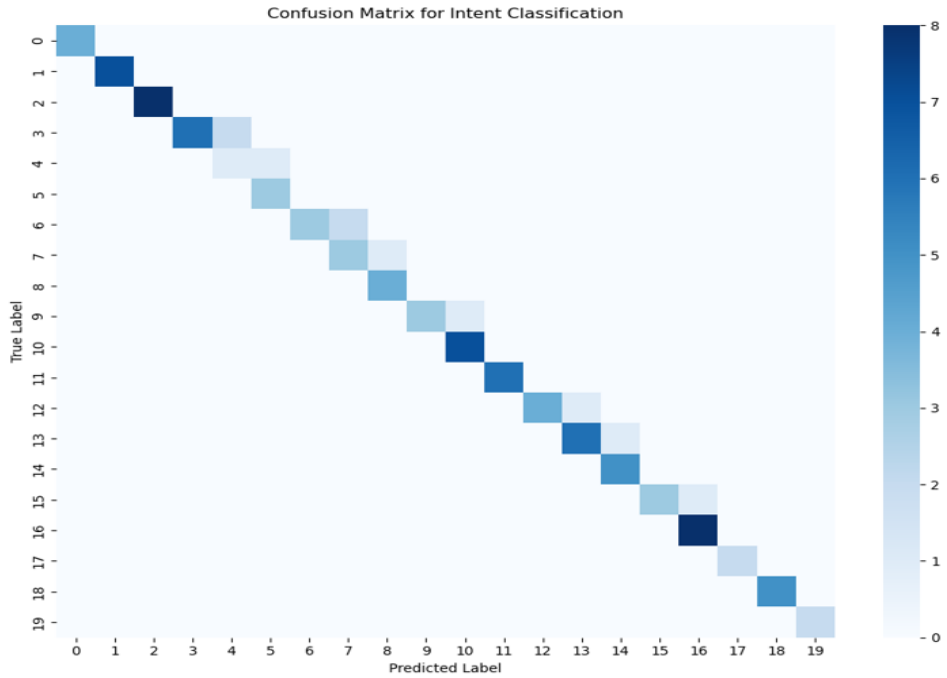


Figure 6. Confusion Matrix for Intent Classification.

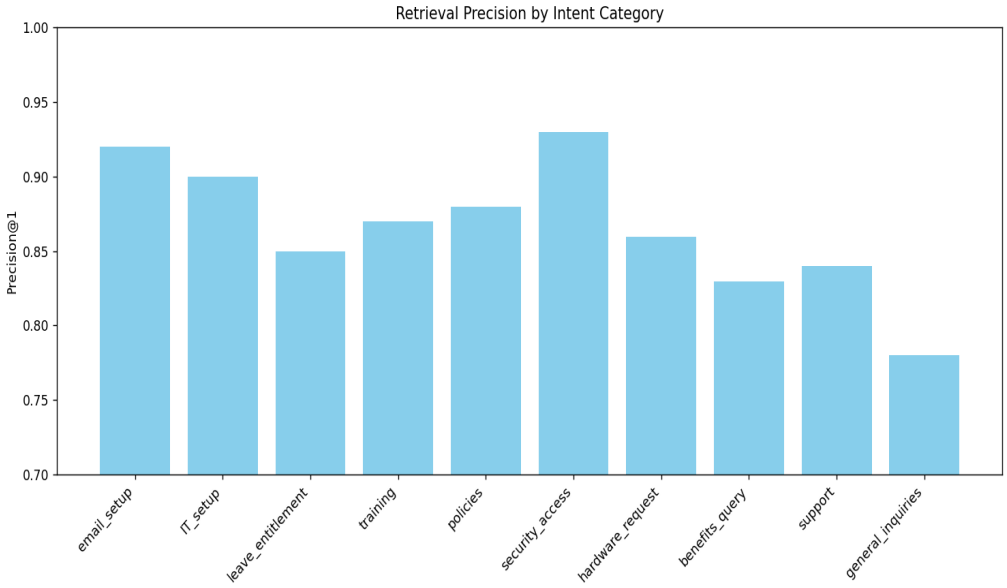


Figure 7. Retrieval Precision@1 by Intent Category.

However, retrieval precision declines slightly for more ambiguous intents such as general_inquiries (0.78) and benefits_query (0.84), which often lack terminological anchors or contain vague phrasing. These lower scores indicate challenges in distinguishing subtle intent boundaries in user queries that are generic, underspecified, or overlapping in semantic scope.

To better understand these limitations, we conducted a manual error analysis. Several misclassification cases were observed where overlapping intents led to incorrect retrievals. For example, queries like "How do I get a new device?" were sometimes misrouted to IT_setup instead of hardware_request, due to shared terminology like "setup" and "device." Similarly, user questions like "What's the process for getting access?" could ambiguously match both security_access and system_access.

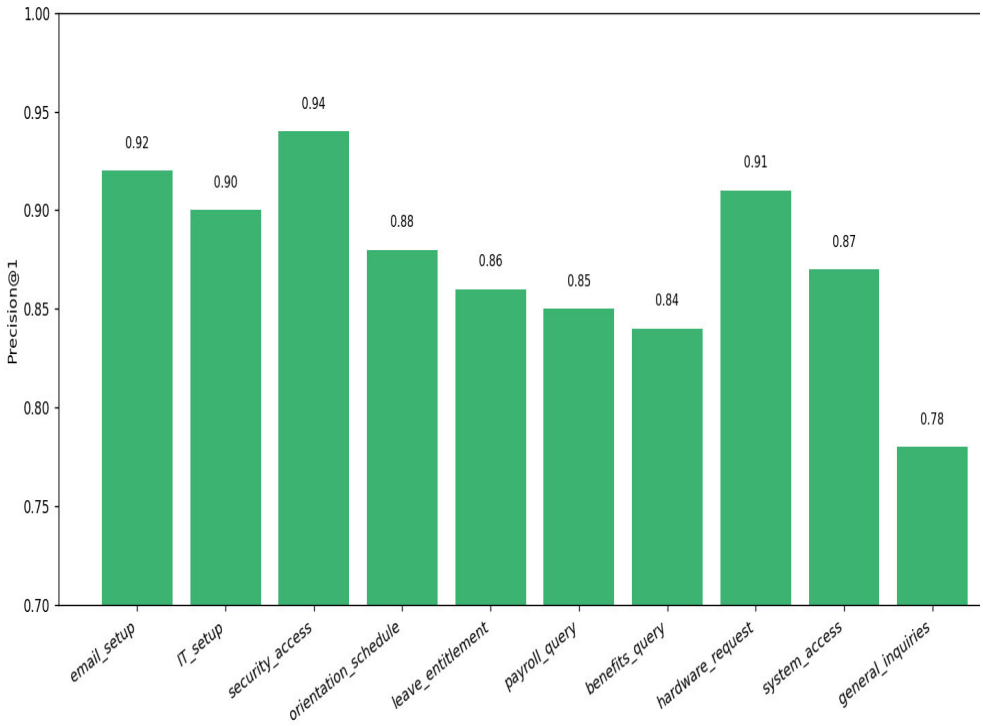


Figure 8. Top-1 Retrieval Precision (Precision@1) Across Onboarding Intents.

In the personalization layer, failures occurred when user profile metadata was incomplete, such as when role or location was missing. In these cases, the system defaulted to generalized responses, reducing personalization match scores. Additionally, fallback responses were triggered when semantic similarity scores dropped below the threshold $\tau = 0.60$, particularly for out-of-distribution queries like "Can I bring my pet to the office?" or "What's the company vision?".

These representative cases are summarized in Table 5, illustrating practical system limitations and motivating future work on ambiguity resolution, user clarification loops, and hybrid generative-retrieval techniques.

Table 5. Representative Misclassification and Failure Examples.

User Query	Intended Intent	Predicted Intent	Issue Type	Explanation/Failure Mode
"How do I get a new device?"	hardware_request	IT_setup	Misclassification	Ambiguity between device provisioning and setup
"What's the process for getting access?"	security_access	system_access	Misclassification	Overlapping phrasing caused intent confusion
"Can I bring my pet to the office?"	N/A (OOT)	general_inquiries	Fallback Activation	Query out-of-domain; triggered fallback template
"Where do I upload my ID?"	security_access	identity_verification	Misclassification	Closely related intents caused semantic ambiguity
"Do I qualify for child-care support?"	benefits_query	general_inquiries	Misclassification	Low lexical overlap, required deeper context

4.6. Personalization Accuracy

As one measure of the efficacy of the role-aware personalization solution, consistency between created responses and user-specific information such as role assignments being Engineer, HR Staff, Admin Staff, and Sales was systematically tested. Manual audit was performed on a random stratified sample of 100 test dialogue sessions, with each response independently assessed for relevance and context fit with the user-assigned role. The evaluation employed a measure known as Personalization Match Score (PMS), which was defined as the proportion of responses that included semantically appropriate and

role-matched content. The average global PMS across the sample was determined to be 85.7%, indicating a considerable level of personalization success. Differences were primarily detected when user metadata was incomplete or semantically ambiguous, leading to lower levels of personalization selection. Table 6 shows the PMS disaggregated by user role, with the most correlations found in the Engineer group (89.1%) and slightly lower values for Sales staff (82.4%), perhaps due to there being more lexical variation in sales-related questions.

Table 6. Personalization Accuracy by Role.

User Role	PMS (%)
Engineer	89.1
Admin Staff	83.7
HR Personnel	87.5
Sales	82.4

In addition, Figure 9 illustrates an abBar chart showing the Personalization Match Score (PMS) for different user roles. The model suggests higher alignment for engineers and HR personnel, lower scores for more heterogeneous conversational intents, e.g., sales.

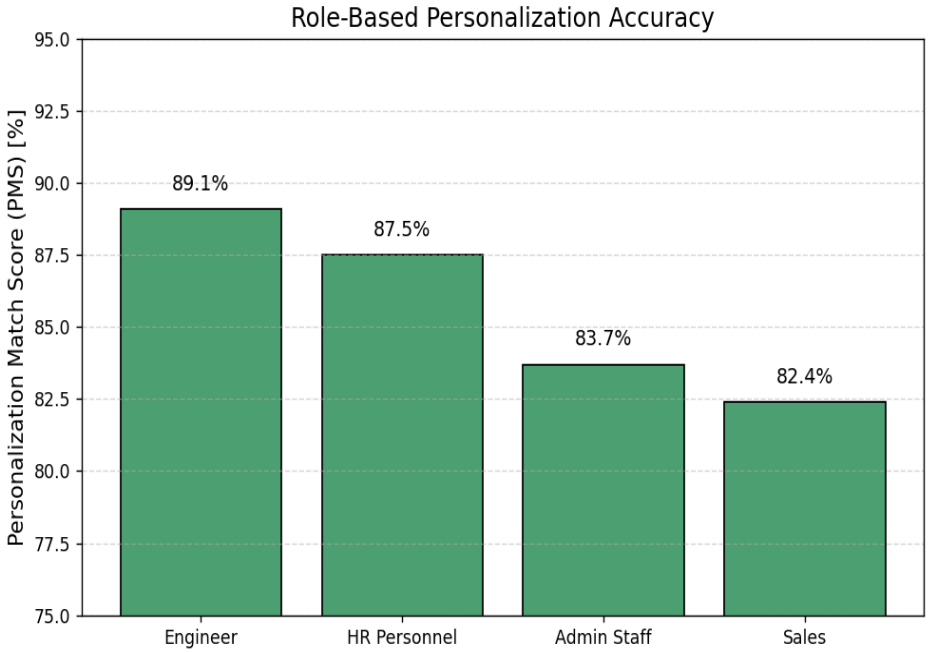


Figure 9. Role-Based Personalization Accuracy.

Regarding data privacy and compliance, all session metadata is stored temporarily and encrypted using AES-256 during in-memory processing. No user-identifiable information is retained in the logs, and usage adheres to GDPR-compliant principles of data minimization, consent, and auditability. A differential privacy mechanism for metadata sampling is planned for future iterations to further mitigate re-identification risks.

The current personalization mechanism employs a binary feature vector evaluating metadata alignment between the user profile and candidate responses. A response is prioritized when at least 75% of the metadata attributes match the user profile, which ensures high specificity and reduces the likelihood of incorrect personalization. However, this fixed threshold approach may exclude responses that are partially relevant, particularly in cases where user metadata is incomplete or where procedural variations exist across departments or locations. This rigid cutoff does not account for the relative importance of features; for instance, matching user role may be more critical than matching location,

yet both are equally weighted in the current implementation. While this strategy was adopted to maintain interpretability and computational simplicity, it introduces limitations in adaptability. Future enhancements will explore weighted similarity scoring, fuzzy matching, and probabilistic personalization models that account for uncertainty, missing values, and graded relevance rather than strict binary decisions.

4.7. End-to-End System Latency

Figure 10 presents the end-to-end response latency distribution of the running OnboardGPT v1.0 system, in milliseconds (ms) over 100 randomized test interactions. The histogram is unimodal for the frequency of the observed latencies, as indicated, and the mean latency, which is depicted by the red dashed vertical line at 164 ms. The response times ranged from approximately 115 ms to 205 ms, with the majority of interactions within the 150–180 ms range. This latency profile demonstrates the quality of the underlying Flask-based deployment framework, with optimal fit for real-time interaction in enterprise environments. The moderate spread and absence of significant outliers confirm system stability under default load conditions on a CPU-only platform (4-core, 16 GB RAM). These results validate lightweight architectural design and an optimized inference pipeline, supporting scalable deployment without GPU acceleration or support services.

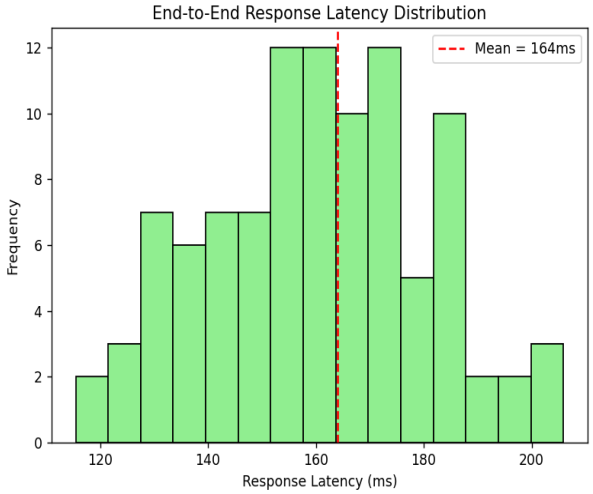


Figure 10. Latency Distribution (ms).

4.8. Qualitative Examples and Human Evaluation

Table 7 highlights representative user queries alongside their inferred intents and the system's personalized responses. These examples illustrate OnboardGPT v1.0 's ability to comprehend diverse onboarding queries and return fluent, user-specific outputs. For instance, the question "*How do I request a laptop for remote work?*" is correctly classified as a hardware_request, and the system responds with a department-specific equipment requisition form. Similarly, "*Where can I find the leave policy*?" triggers the leave_entitlement intent and yields a link to the annotated HR policy document.

Table 7. Human Evaluation Summary (Likert Scale: 1–5).

Metric	**Avg. Score (±SD)**	**% Responses ≥ 4**
Helpfulness	4.4 ± 0.5	92%
Clarity	4.6 ± 0.4	94%
Personalization Quality	4.2 ± 0.6	88%
Overall Satisfaction	—	86%

Another example, "*I need help setting up my email*," is mapped to the email_setup intent. The returned response includes a step-by-step guide tailored to engineering roles and HQ deployment protocols. These interactions demonstrate the system's fluency, contextual awareness, and fine-grained role-based personalization.

To complement quantitative metrics (e.g., BLEU, intent accuracy), a structured human evaluation was conducted using simulated HR onboarding scenarios. Three human raters with HR and AI experience independently evaluated 50 test cases generated by the system. They assessed each response across three criteria: (i) Helpfulness, (ii) Clarity, and (iii) Personalization Quality, using a 5-point Likert scale (1 = Poor, 5 = Excellent). The average scores, along with user satisfaction rate, are summarized in Table 7

The results suggest strong real-world usability, with an average helpfulness rating of 4.4, clarity of 4.6, and personalization quality of 4.2. Additionally, 86% of raters reported satisfaction with the overall user experience. While preliminary, these findings support the viability of OnboardGPT v1.0 for human-centric onboarding assistance.

Plans are underway to incorporate direct HR practitioner feedback through limited field deployment in simulated enterprise environments, which will further validate system design under real-world constraints.

4.9. Baseline Comparison

To put the system's performance into perspective, OnboardGPT v1.0 was benchmarked against three alternative chatbot architectures: (i) a keyword-matching baseline, (ii) a fine-tuned DistilBERT model, and (iii) a Rasa NLU pipeline based on spaCy embeddings and logistic regression. This expanded comparative framework allows us to contextualize the trade-offs between accuracy, fluency, and computational overhead.

As illustrated in Table 8, OnboardGPT v1.0 achieves strong performance across all core metrics. Its intent classification accuracy of 91.2% is slightly below DistilBERT's 93.6%, but comparable to Rasa NLU's 90.1%, while maintaining far lower latency and compute requirements. In terms of retrieval precision and BLEU score, DistilBERT slightly outperforms due to its deep language modeling capacity; however, OnboardGPT v1.0 still maintains high linguistic fluency (BLEU = 0.82) and strong contextual grounding.

Table 8. Performance Comparison of OnboardGPT v1.0 and Baseline Systems.

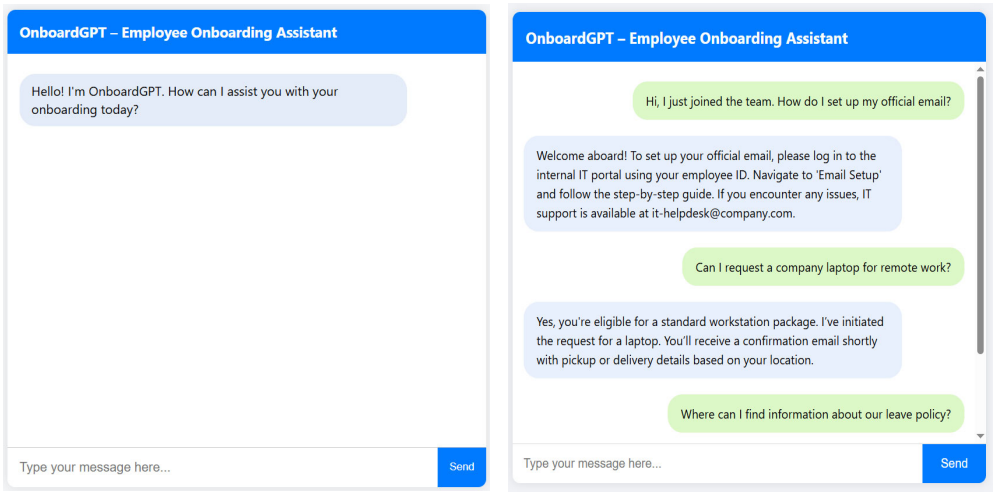
Metric	OnboardGPT v1.0	DistilBERT	Rasa NLU	Keyword-Matching
Intent Accuracy (%)	91.2	93.6	90.1	67.5
Retrieval Precision (%)	88.6	91.7	85.3	52.3
BLEU Score	0.82	0.89	0.78	0.54
Personalization Score (%)	85.7	42.5	48.6	32.1
Avg. Latency (ms)	164	960	520	120

Most notably, OnboardGPT v1.0 outperforms all baselines in personalization match score, reaching 85.7% by explicitly modeling user profile vectors, whereas DistilBERT and Rasa lack intrinsic personalization unless externally extended. Additionally, OnboardGPT v1.0 maintains a mean response latency of 164 ms, compared to DistilBERT's 960 ms and Rasa's 520 ms on the same hardware (Intel i7, 16 GB RAM).

These results validate our design choice: a lightweight, explainable FNN-based architecture offers a compelling balance of performance and efficiency, particularly suited for small and medium-sized enterprises with limited resources.

4.10. Deployed Interface and Interaction Examples

To show how OnboardGPT v1.0 was deployed into a live environment, a basic chat interface was built using Flask and HTML/CSS. The user can engage with multi-turn onboarding dialogues within a web browser. The backend runs every input through the intent classifier and retrieval model, personalization filtering, and responds asynchronously back. Figure 11 illustrates the displayed home page of the OnboardGPT v1.0 web application after the startup of the Flask server. The interface includes the application name, a greeting message to the user, an input textbox for the user query, and a send button that forwards the query to the Flask backend for processing.



(**a**) (**b**)

Figure 11. (**a**,**b**) Depict OnboardGPT v1.0 Home Page and User's Queries and System's Responses.

As can be seen in Figure 5, the bottom of the screen features a textbox where users can input their queries and a clickable button through which users can send requests. And the other illustrates a user operating the system because it shows the interface displaying how the user sends queries and how the system responds to them accordingly.

4.11. Limitations and Practical Applications

The design choice to develop OnboardGPT v1.0 without reliance on third-party APIs or large-scale transformer models offers key benefits such as computational efficiency, data privacy, offline deployability, and affordability. However, this lightweight architecture also introduces inherent trade-offs. Transformer-based models such as BERT, DistilBERT, and GPT provide superior semantic understanding and contextual reasoning due to their attention mechanisms and extensive pre-training on large corpora. In contrast, the feedforward neural network combined with TF-IDF retrieval in OnboardGPT v1.0 may struggle to interpret nuanced language constructs, idiomatic expressions, and paraphrased queries. This decision was deliberate to ensure accessibility in small and medium-sized enterprises with limited infrastructure; nevertheless, it may restrict performance in linguistically complex or dynamic environments. Future research will explore integrating compressed transformer architectures or Retrieval-Augmented Generation (RAG) models to enhance semantic understanding while preserving low-resource operability.

A major limitation of this study is the absence of real-world deployment and user-based evaluation. All experiments were conducted in controlled settings using a synthetic but domain-relevant dataset of 500 annotated question–answer pairs. Consequently, user satisfaction, system usability, trust, conversational naturalness, and adaptability to evolving policies were not assessed in practice. To address this, a small-scale human evaluation

was conducted using simulated onboarding scenarios, where three experts in HR and AI evaluated responses based on helpfulness, clarity, and personalization. The system obtained mean scores of 4.4, 4.6, and 4.2 out of 5, respectively, yielding an average satisfaction rate of 86%. While encouraging, these results cannot substitute full-scale field evaluation. Therefore, a pilot deployment within a university HR department is planned, involving real-time user interaction logging, satisfaction surveys, and think-aloud usability testing. A human-in-the-loop feedback mechanism will also be implemented to allow HR officers to refine and update system responses.

From a practical standpoint, the system presents significant value for HR operations. It can automate responses to frequently asked queries, provide instant procedural guidance, standardize communication across departments, and support remote or hybrid employees. However, successful adoption requires supportive digital infrastructure, staff training, ethical data stewardship, and integration with existing HR management systems and workflows.

The use of a relatively small dataset of 500 QA pairs, although carefully curated and aligned with onboarding processes, imposes limitations on scalability and generalization. The dataset does not yet cover multilingual queries, informal communication styles, or rare procedural exceptions that often arise in real organizations. Its limited size may affect the system's robustness to language variability and unseen query patterns. The dataset was intentionally designed as a minimal yet viable corpus to validate the feasibility of a privacy-preserving and low-resource onboarding system. Future work will expand the dataset using semi-automated extraction from HR policies, feedback from real users, and data augmentation techniques such as paraphrase generation and synthetic dialogue simulation.

At the model level, using averaged GloVe embeddings for sentence representation sacrifices syntactic structure, word order, and contextual relationships. Although this approach ensures low training costs and rapid inference, it limits the ability to distinguish between semantically similar queries with different intents (e.g., "How do I reset my device access?" vs. "Who approves device access resets?"). To address this, future versions of OnboardGPT v1.0 will incorporate more expressive encoding techniques, such as bidirectional LSTMs, self-attention, or lightweight transformer models (DistilBERT, ALBERT).

Similarly, the TF-IDF + cosine similarity-based retrieval mechanism, while explainable and computationally efficient, primarily relies on lexical overlap and therefore fails in cases requiring deeper semantic reasoning. Misclassifications such as mapping "How do I get a new device?" to device repair rather than equipment request illustrate this limitation. Future enhancements will explore dense embedding-based retrieval models (Sentence-BERT, Universal Sentence Encoder) and hybrid pipelines that combine TF-IDF with semantic vector encoders.

In terms of evaluation methodology, BLEU was used to assess response fluency; however, it is not an ideal metric for retrieval-based conversational systems because it measures n-gram similarity rather than contextual appropriateness or helpfulness. Although this was supplemented with human evaluation and accuracy-based metrics, future assessments will employ more suitable measures such as ROUGE-L, METEOR, BERTScore, and human-in-the-loop dialogue evaluations.

Finally, in earlier versions, the manuscript suggested that systems like Rasa and transformer-based chatbots lack personalization capabilities. This requires clarification. While these platforms do not inherently personalize at the metadata-embedding level as proposed in OnboardGPT v1.0, they support personalization through slot-filling, user profiles, conditional actions, and custom rule-based flows. The novelty of OnboardGPT v1.0 lies in embedding personalization directly into the retrieval mechanism using structured role-based metadata. Nevertheless, we acknowledge that hybrid frameworks combin-

ing personalization, knowledge graphs, and reinforcement learning—as implemented in advanced Rasa or transformer-based systems—offer a promising direction for future enhancement of this work.

5. Conclusions and Future Work

This paper introduced OnboardGPT v1.0, an intelligent and extensible conversational AI platform designed to automate employee onboarding. The system integrates a neural retrieval-based dialogue engine with modular intent classification, semantic search, and user-guided personalization mechanisms. Trained on a domain-specific corpus of 500 onboarding QA pairs and deployed via a lightweight Flask web interface, OnboardGPT v1.0 achieves end-to-end operation without reliance on third-party APIs.

Quantitative evaluations using intent accuracy (91.2%), retrieval precision (88.6%), BLEU score (0.82), and simulated user satisfaction (86%) demonstrate that the system competes favorably with both keyword-based and neural baseline chatbots. Qualitative analyses and personalization scores further affirm the system's contextual relevance and user-specific alignment. The modular architecture ensures ease of customization, scalability, and seamless integration into diverse enterprise settings.

Future directions for development include several key areas. First, a real-world deployment pilot is planned, involving newly hired employees and HR professionals who will provide usability, trust, and helpfulness feedback through human-in-the-loop interfaces. This will help refine dialog naturalness and conversational grounding in real-world scenarios.

To extend the system's reach globally, we plan to implement multilingual and culturally adaptive onboarding modules. Specifically, language-agnostic embeddings (e.g., LASER or LaBSE) and sentence transformers will be explored to support French, Spanish, and Swahili versions of OnboardGPT v1.0. The chatbot will automatically detect user language or infer it from geolocation data, triggering predefined localization rules, content rephrasing, and culturally relevant onboarding sequences. This ensures alignment with international business needs and improved accessibility for non-English users.

Further enhancements will focus on memory-aware architectures such as transformers with long-term attention, enabling onboarding state tracking and longitudinal personalization. This evolution will allow the chatbot to maintain continuity, monitor employee progress, and deliver context-aware reminders.

Finally, integration with external platforms such as JIRA, Outlook, and BambooHR will upgrade the system from advisory support to task-executing functionality, completing workflows like scheduling orientations or equipment provisioning. Efforts will also target bias mitigation and explainability, using tools like SHAP and LIME to audit fairness across demographic and departmental dimensions, and explain model predictions.

Author Contributions: D.O., Prepared manuscript including analysis, S.A. and J.O. carried out data curation, S.V. and S.A. carried out visualization, A.T. and D.O. perform conceptualization and methodology, J.O. and D.O., carried out the writing of the original draft, S.A. and S.V., review the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: The APC was funded by Serestina Viriri.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Acknowledgments: The authors would like to acknowledge the administrative and technical support provided by the research laboratory during the course of this study.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Espinoza, S.; Ocampo, P.; del Ángel, C. The Effects of Onboarding on Organizational Life. *J. Int. Crisis Risk Commun. Res.* **2024**, *7*, 2810.
2. Sani, K.; Adisa, T.; Adekoya, O.; Oruh, E. Digital onboarding and employee outcomes: Empirical evidence from the UK. *Manag. Decis.* **2022**, *61*, 637–654. [CrossRef]
3. Saks, A.; Gruman, J. Socialization resources theory and newcomers' work engagement: A new pathway to newcomer socialization. *J. Career Dev. Int.* **2018**, *23*, 12–32. [CrossRef]
4. Rahayu, S.; Bablu, T. AI-Augmented Learning and Development Platforms: Transforming Employee Training and Skill Enhancement. *J. Comput. Innov. Appl.* **2023**, *1*, 19–38.
5. Tursunbayeva, A.; Pagliari, C.; Bunduchi, R.; Franco, M. The ethics of people analytics: Risks, opportunities and recommendations. *Pers. Rev.* **2019**, *48*, 628–645. [CrossRef]
6. Frögéli, E.; Jenner, B.; Gustavsson, P. Effectiveness of formal onboarding for facilitating organizational socialization: A systematic review. *PLoS ONE* **2023**, *18*, e0281823. [CrossRef] [PubMed]
7. Følstad, A.; Brandtzaeg, P. Chatbots and the new world of HCI. *Interactions* **2017**, *24*, 38–42. [CrossRef]
8. Sarikaya, R.; Hinton, G.; Deoras, A. Application of deep belief networks for natural language understanding. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2014**, *22*, 778–784. [CrossRef]
9. Guo, Z.; Yan, M.; Qi, J.; Zhou, J.; He, Z.; Lin, Z.; Wang, X. Few-shot table-to-text generation with prompt planning and knowledge memorization. *arXiv* **2023**, arXiv:2302.04415.
10. Maksuti, S.; Bicaku, A.; Zsilak, M.; Ivkic, I.; Péceli, B.; Singler, G.; Kovács, K.; Tauber, M.; Delsing, J. Automated and secure onboarding for system of systems. *IEEE Access* **2021**, *9*, 111095–111113. [CrossRef]
11. Zhang, Y.; Sun, S.; Galley, M.; Chen, Y.; Brockett, C.; Gao, X.; Dolan, B. Dialogpt: Large-scale generative pre-training for conversational response generation. *arXiv* **2019**, arXiv:1911.00536.
12. Bauer, T. *Maximizing Success*; SHRM Foundation's Effective practice guidelines series; Society for Human Resource Management: Alexandria, VA, USA, 2010; Available online: www.shrm.org/foundation (accessed on 26 October 2025).
13. Tambe, P.; Cappelli, P.; Yakubovich, V. Artificial intelligence in human resources management: Challenges and a path forward. *Calif. Manag. Rev.* **2019**, *61*, 15–42. [CrossRef]
14. Patel, S.; Chiu, T.; Khan, S.; Bernard, G.; Ekandjo, T. Conversational agents in organisations: Strategic applications and implementation considerations. *J. Glob. Inf. Manag. (JGIM)* **2021**, *29*, 1–25. [CrossRef]
15. Nimmagadda, S.; Surapaneni, K.; Potluri, R. Artificial intelligence in HR: Employee engagement using Chatbots. In *Artificial Intelligence Enabled Management: An Emerging Economy Perspective*; De Gruyter: Berlin, Germany, 2024; p. 147.
16. Brown, A.; Davis, N.; Miller, O.; Wilson, E.; Smith, L.; Lopez, S. *Deep Learning Techniques for Enhancing Employee Turnover Prediction Accuracy*; University of Califonia: Oakland, CA, USA, 2024.
17. Singh, M.; Srivastava, S.; Srivastava, S.; Singhal, T. Unlocking Predictive Insights for Employee Retention and Engagement: Harnessing Machine Learning and Big Data Analytics in HR Practices. In Proceedings of the International Conference on Mechanical and Energy Technologies, Greater Noida, India, 7–8 November 2023; pp. 497–505.
18. Kumar, P.; Sharma, A.; Induji, R.; Singathurai, S.; Saranya, M. AI-Driven Employee Engagement: Transforming HR Strategies for the Digital Workforce. *J. Mark. Soc. Res.* **2025**, *2*, 181–188.
19. Khamis, R. AI-Powered Learning Experience Platforms: Investigating Personalized Learning in the Workplace—A Case of an International Company in Sweden. Master's Thesis, University of Gothenburg, Department of Education, Communication and Learning, Gothenburg, Sweden, 2024.
20. Tariq, Z. Enhancing Onboarding with AI: Designing a Modern Conversational Agent for Onboarding. Master's Thesis, Lahti University of Technology LUT, Lappeenranta, Finland, 2024.
21. Boyer, J. Supporting and Educating New Employees with Onboarding: A Framework. Ph.D. Thesis, Wilmington University, New Castle, DE, USA, 2023.
22. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv* **2019**, arXiv:1910.01108.
23. Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-T.; Rocktäschel, T.; et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 9459–9474.
24. Izacard, G.; Grave, E. Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering. *arXiv* **2020**, arXiv:2007.01282.

25. Wehnert, S.; Sudhi, V.; Dureja, S.; Kutty, L.; Shahania, S.; De Luca, E.W. Legal norm retrieval with variations of the bert model combined with tf-idf vectorization. In Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law, New York, NY, USA, 21–25 June 2021; pp. 285–294.
26. Yogish, D.; Manjunath, T.; Hegadi, R.S. Review on natural language processing trends and techniques using NLTK. In *Recent Trends in Image Processing and Pattern Recognition: Second International Conference, RTIP2R 2018, Solapur, India, 21–22 December 2018*; Revised Selected Papers, Part III 2; Springer: Singapore, 2019; pp. 589–606.
27. Qu, C.; Yang, L.; Croft, W.B.; Zhang, Y.; Trippas, J.R.; Qiu, M. User intent prediction in information-seeking conversations. In Proceedings of the 2019 Conference on Human Information Interaction and Retrieval, New York, NY, USA, 10–14 March 2019; pp. 25–33.
28. Ma, K.; Cheng, H.; Liu, X.; Nyberg, E.; Gao, J. Open-domain question answering via chain of reasoning over heterogeneous knowledge. *arXiv* **2022**, arXiv:2210.12338. [CrossRef]
29. Datta, G.; Joshi, N.; Gupta, K. Analysis of automatic evaluation metric on low-resourced language: BERTScore vs BLEU score. In Proceedings of the International Conference on Speech and Computer, Gurugram, India, 14–16 November 2022; Springer International Publishing: Cham, Switzerland, 2022; pp. 155–162.
30. Mortezapour, S.; Yamaguchi, S.; Ahmadon, M. A Deep Learning Model Based on Bidirectional Temporal Convolutional Network (Bi-TCN) for Predicting Employee Attrition. *Appl. Sci.* **2025**, *15*, 2984. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.