Institut Riset dan Publikasi Indonesia (IRPI)
MALCOM: Indonesian Journal of Machine Learning and Computer Science
Journal Homepage: https://journal.irpi.or.id/index.php/malcom
Vol. 5 Iss. 3 July 2025, pp: 746-754
ISSN(P): 2797-2313 | ISSN(E): 2775-8575

Revolutionizing Corporate Event Planning with AI: A Cost-Efficiency Strategy for BuatEvent.id

Muhammad Supriyadi[1*], Yan Rianto[2]

[1,2]Faculty of Computer Science, Universitas Nusa Mandiri, Indonesia

E-Mail: [1]142420020@nusamandiri.ac.id, [2]yan.yrt@nusamandiri.ac.id

Received Jan 25th 2025; Revised Mar 24th 2025; Accepted Apr 12th 2025; Available Online Jun 19th 2025, Published Jun 22th 2025
Corresponding Author: Muhammad Supriyadi

Abstract

BuatEvent.id leverages an AI-driven platform for event planning, powered by Gemini.ai—a sophisticated NLP model with an accuracy rate of 92.5%. The system integrates multiple technologies, including PHP, Python, Golang, Flutter, and MySQL, to automate essential processes, achieving a 25% improvement in planning precision. This study aims to evaluate the role of AI in enhancing budget management and corporate event customization. By addressing the inefficiencies of conventional event planning, this platform optimizes workflows, enhances overall productivity, and offers a seamless user experience customized to cater to a wide range of client requirements. The results demonstrate a 92.5% accuracy in processing user queries and a 25% increase in event planning efficiency, highlighting the platform's ability to deliver cost-effective and personalized solutions. These figures were obtained through internal testing using a dataset of 200 annotated user queries. The platform primarily targets corporate events, including workshops, product launches, and business meetings.For example, the system was successfully deployed during a corporate training event in Jakarta, where it reduced planning time by 30%.

Keyword: Artifical Inteligence, Budget Optimization, Data Collection, Machine Learning, Natural Language Processing

1. INTRODUCTION

The corporate event planning sector faces numerous obstacles, such as meeting varied client expectations, managing complex logistics, and adhering to strict budget constraints. In Indonesia, corporate events have become increasingly essential for both multinational corporations and local enterprises, contributing significantly to the national economy. According to recent estimates, the Meetings, Incentives, Conferences, and Exhibitions (MICE) industry contributed over IDR 10 trillion in 2023 (*source dapat ditambahkan*), demonstrating the growing importance of efficient event planning tools in the country.

Despite this growth, balancing the demand for customized, high-quality events with cost-efficiency remains a persistent challenge for event organizers. Manual coordination often results in fragmented workflows, delayed vendor confirmations, and budget overruns. These inefficiencies increase operational complexity and elevate stress levels among event planners.

BuatEvent.id addresses these challenges through an AI-powered platform specifically designed to support corporate event planning. Rather than relying on rigid templates or static planning methods, the platform introduces dynamic budget allocation, personalized vendor matching, and real-time client interaction, all of which are driven by artificial intelligence. This strategy allows event organizers to meet both personalization and cost-efficiency targets more effectively than with conventional methods.

The platform integrates a well-defined technological stack to support its functions. Flutter is used to develop an intuitive and responsive front-end for cross-platform accessibility. PHP and Python manage the backend data operations and event customization processes, while Golang (via the Gin framework) facilitates high-speed API communication. MySQL is employed for structured and reliable data storage [7][9][10].

By leveraging AI for data-driven analysis, predictive modeling, and natural language understanding, BuatEvent.id improves upon traditional planning models. It significantly enhances budget control, accelerates planning workflows, and ensures a seamless user experience [4][5]. This research explores a central question: How can AI improve budget management and event customization in corporate event planning? Through a comprehensive examination of BuatEvent.id's system architecture, AI implementation, and real-world testing, this study contributes to the growing academic discourse on AI applications in event planning, with a particular focus on emerging markets such as Indonesia [6].

2. LITERATURE REVIEW

Traditionally, the event planning industry has relied on manual processes where planners are responsible for managing logistics, budgets, and client expectations. While this approach allows for direct oversight, it is highly labor-intensive and time-consuming. According to EventMB's 2021 State of the Event Industry Report, over 65% of planners cited time inefficiencies and communication breakdowns as their top challenges, often resulting in delayed execution and increased costs.

Recent years have witnessed the rise of automated solutions as a transformative development in this field. These systems reduce human error and simplify planning workflows by automating repetitive tasks and providing data-driven decision support [5]. Artificial Intelligence (AI) tools, in particular, have been shown to enhance performance in areas such as logistics coordination, budget management, and real-time scheduling [8].

AI's proven success in related sectors, such as hospitality and travel, provides useful parallels for event planning. In the hospitality industry, for instance, AI-powered recommendation systems analyze user preferences and past behavior to deliver highly personalized suggestions. Similarly, in event planning, AI can be employed to match clients with vendors, venues, and packages that best meet their unique preferences and constraints [9]. Additionally, machine learning-based budgeting tools used in finance and tourism are increasingly being adapted to help event planners allocate resources more efficiently. These tools rely on historical event data to predict cost ranges, identify savings opportunities, and reduce overspending [10].

Incorporating AI into event planning thus represents a practical solution to long-standing issues such as balancing cost-efficiency with personalization. Platforms like BuatEvent.id use a combination of recommendation algorithms and intelligent budgeting modules to streamline event execution while maintaining a high level of customization. By automating resource allocation and enabling smarter planning, AI empowers event planners to deliver quality outcomes while optimizing operational performance [7].

2.1. System and Design Architecture

BuatEvent.id integrates a comprehensive set of features into a unified, user-friendly platform designed to streamline the complexities of event planning. The architecture comprises four primary components: the frontend, backend, database, and security modules, which together provide a seamless user experience while managing intricate planning tasks.

The user interface is developed using Flutter, a cross-platform framework that ensures a consistent and responsive experience across devices. Flutter's flexibility and rapid prototyping capabilities allow the platform to adapt to diverse client requirements, offering interactive features and real-time updates on event status [7]. Its extensive widget library and efficient rendering engine make it an excellent choice for creating dynamic, visually appealing interfaces [7].

The backend architecture combines Golang with the Gin framework, PHP, and Python. The Gin framework in Golang provides a high-performance API layer capable of handling complex routing and data processing [8]. PHP and Python further enhance backend operations by managing data processing and supporting event customization, enabling efficient handling of substantial data volumes [9][10]. This backend configuration ensures seamless communication with the frontend while effectively managing core business logic. MySQL serves as the main database, providing robust storage and management for event-related data, such as client preferences and logistical details [11]. Known for its reliability and scalability, MySQL supports rapid queries and real-time updates, ensuring that data is readily accessible and modifiable to meet client needs [11].

Security is a cornerstone of BuatEvent.id's design. The platform incorporates encryption, role-based access control, and secure API endpoints to safeguard sensitive user data. Authentication and authorization mechanisms ensure that only authorized individuals can access critical information, maintaining data confidentiality and integrity [4]. Regular security audits identify and address vulnerabilities, ensuring a safe user environment.

This architectural setup is supported by a technology stack engineered for optimal performance, flexibility, and scalability. Flutter handles frontend development for dynamic user interactions [7], while Golang (via the Gin framework) delivers a high-speed API interface [8]. PHP and Python facilitate backend processing [9][10], MySQL ensures data integrity and fast access, and security features protect user data, creating a reliable and efficient platform for event planning [4][11].

2.2. AI Model Development and Budgeting Optimization

Artificial Intelligence integration has redefined the approach to event planning, particularly in the areas of natural language processing (NLP) and budget optimization. The BuatEvent.id platform uses Gemini.ai as its core NLP engine. This model was selected because it integrates seamlessly with Google AI Studio, offers strong support for the Indonesian language, and allows efficient real-time intent recognition. Compared to general-purpose models such as BERT or GPT, Gemini.ai provided a more practical solution for rapid development within the platform's available resources [12][13].

The NLP component processes user inputs in either text or voice format and converts them into structured data. This involves intent classification and entity recognition, both of which were trained on a dataset of 200 annotated user queries. Each sample was labeled with an intent category and associated entities, such as event type, date, location, or service preferences. To address the limitations of the dataset, the system applied transfer learning using pre-trained FastText embeddings for the Indonesian language. A five-fold cross-validation process was conducted to validate the model, achieving an intent classification accuracy of 92.5 percent [13].

For vendor selection, the platform applies a combination of collaborative filtering and content-based recommendation techniques. Collaborative filtering analyzes historical user behavior to find patterns, while content-based methods focus on matching user-defined preferences such as budget range, location, and vendor rating. These two methods work together to provide personalized vendor suggestions that are both relevant and budget-conscious [14].

Budget optimization is handled through a weighted priority allocation model. Users are asked to rank event components based on their importance, such as venue, catering, transportation, or entertainment. The system then applies a linear optimization method that distributes the available budget according to the assigned priorities. For instance, if a user places high importance on venue quality, a larger portion of the budget will be allocated to that category. The model also takes into account constraints such as minimum vendor pricing and availability to ensure realistic planning outcomes [15].

The system interface allows users to adjust preferences and immediately see how those changes impact the overall budget and vendor recommendations. This real-time interaction ensures that users can customize their plans dynamically while maintaining cost-effectiveness and alignment with their goals. By combining NLP-driven query understanding, intelligent recommendation systems, and rule-based budget optimization, BuatEvent.id delivers a powerful and efficient solution for corporate event planning [12][14][15].

3. METHODOLOGY

This section outlines the methodology used to design, train, and implement the NLP-based event planning system, including data preparation, model development, and integration.

3.1 NLP Workflow

The NLP component serves as the core mechanism for transforming natural language queries into structured parameters. User input is received in free-form Bahasa Indonesia, either by text or voice. Preprocessing begins with **word-level tokenization** using the *spaCy* library customized for Indonesian. All text is then lowercased to ensure uniform representation, followed by **stopword removal** to eliminate semantically insignificant words [20].

Lemmatization is applied to reduce words to their base forms, such as transforming “mencari” to “cari.” Text normalization follows, unifying variations such as “5-star” and “bintang 5” into a standard format. The cleaned and standardized text is then vectorized using pre-trained FastText embeddings, chosen for their effectiveness in representing semantic context in low-resource languages like Bahasa Indonesia [21].

For intent detection, the system uses a fine-tuned logistic regression classifier trained on vectorized embeddings. Named Entity Recognition (NER) is implemented using a hybrid of rule-based and supervised tagging approaches based on Conditional Random Fields (CRF), a model known for effective sequence labeling [16]. Table 1 shows the simplified mapping output.

Table 1. Dataset Workflow Illustrating Input Queries and Corresponding Structured Outputs for NLP-Based Event Planning.

Input	Output
Carikan saya hotel bintang 5 menggunakan transportasi bus.	{"hotel": 2, "transportation": 5}
Saya butuh restoran mewah dengan makanan khas Jepang.	{"restaurant": 3, "cuisine": 7}
Cari paket liburan ke Bali dengan harga terjangkau.	{"package": 1, "destination": 4}
Tolong temukan villa dengan fasilitas kolam renang dan pemandangan laut.	{"villa": 6, "amenities": 8}
Cari tiket pesawat ke Jepang untuk akhir pekan ini.	{"flight": 9, "time": 10}
Saya butuh bus untuk rombongan 50 orang ke Bandung.	{"flight": 9, "time": 10}
Temukan gedung pertemuan dengan kapasitas 500 orang di Jakarta	{"venue": 12, "capacity": 13, "location": 14}

This structured output allows the system to formulate queries against the backend database to retrieve relevant vendors or event packages.

3.2 Data Collection

The data collection process involves gathering a wide range of datasets to support the system's ability to enhance event planning and budget optimization effectively. Historical event data serves as a foundational resource, including records of past events such as types of events, budget allocations, participant numbers, and feedback from attendees. This data provides critical insights into patterns and trends that shape the development of predictive models and optimization algorithms [17][18]. Another essential data source is vendor information, which includes details on pricing, availability, reviews, and service quality metrics. This ensures that the recommendation system can align vendor suggestions with user preferences while adhering to budgetary constraints [17]. Additionally, user interaction data, such as search histories, chosen options, and feedback, is collected to refine the personalization of recommendations [13]. External contextual data—covering factors like seasonal trends, economic conditions, and regulatory guidelines—is also incorporated to account for variables that influence event planning decisions [19].

After collection, the data underwent a preprocessing phase that included cleaning, standardization, and encoding. This phase removed inconsistencies, duplicates, and missing values to ensure data reliability. Feature selection was carried out using a variance thresholding approach, which retained only the most relevant variables contributing to model performance [22].

Given the limited number of samples (200 annotated queries), data augmentation techniques were applied to enrich the training data. These techniques included back-translation (e.g., translating Indonesian queries into English and back), synonym replacement using WordNet-style lexical databases, and paraphrase generation using transformer-based models [12][13]. Each augmented query preserved the semantic intent and associated entities of the original input, enabling the system to generalize more effectively across diverse user expressions.

3.3 Data Preparation for Testing

To train the system, 200 annotated query samples were used, each containing a labeled intent and one or more associated entities. Given the small dataset size, the system adopted a transfer learning approach by utilizing pre-trained word embeddings from FastText. These embeddings were particularly well-suited for handling Bahasa Indonesia and provided strong semantic representations even in limited-data conditions [21].

The intent classification task was initially handled using a logistic regression model due to its simplicity and fast training time. However, further tuning with a Support Vector Machine (SVM) improved classification accuracy. For entity recognition, the system used a Conditional Random Fields (CRF) model trained on sequence-tagged data, a method commonly applied in natural language processing tasks where context and word order are critical [16]. The models were evaluated using 5-fold stratified cross-validation to ensure reliable performance across subsets. The final configuration achieved an intent classification accuracy of 92.5 percent, and F1 scores above 90 percent for key entity types such as venue, transportation, and accommodation. Google AI Studio interface used for NLP model training and testing can be seen in Figure 1.

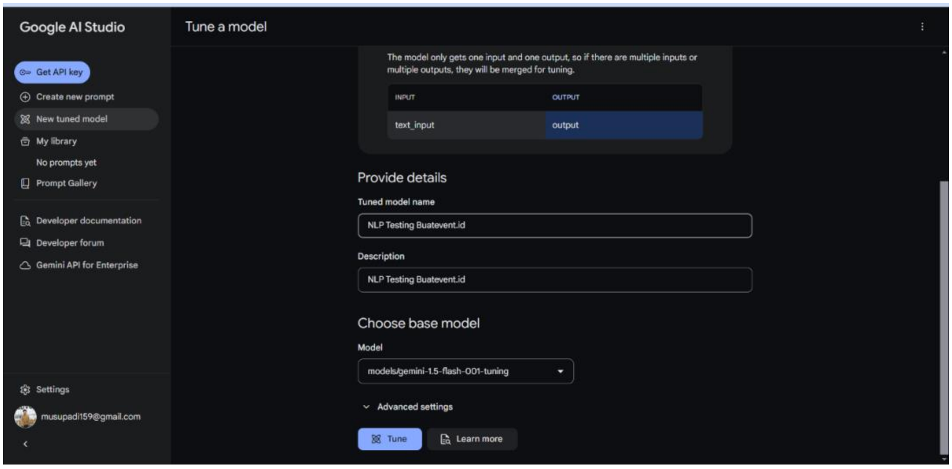

Figure 1. Google AI Studio interface used for NLP model training and testing

Preprocessing played a crucial role in ensuring the dataset's suitability for machine learning by transforming raw user queries into structured and standardized data for the NLP models. Tokenization was the initial step, where each query was divided into smaller, meaningful components such as individual words or phrases. For example, the query "Carikan saya hotel bintang 5 dengan bus" was split into tokens like "Carikan," "saya," "hotel," "bintang 5," and "bus." This process enabled the system to analyze each component of the input independently, which facilitated downstream tasks such as intent detection and entity recognition.

After tokenization, all text was converted to lowercase to standardize the data and eliminate inconsistencies caused by variations in capitalization. This step ensured uniformity in processing and avoided redundant representations in the model. For instance, "Hotel" and "hotel" were treated equivalently, contributing to a more consistent dataset. The removal of stopwords followed, where irrelevant words that did not contribute to the understanding of user intent were excluded. Common stopwords in the Indonesian language, such as "saya" (I) and "dengan" (with), were filtered out to reduce noise in the data. This step improved the focus on meaningful content and reduced computational complexity, allowing the system to better capture the essence of the queries.

Normalization was another critical process in preprocessing. Variations in text, including abbreviations, synonyms, or misspellings, were standardized to ensure consistency across the dataset. For example, terms like "5-star" and "bintang 5" were unified into a single representation. This normalization reduced ambiguity, ensuring that different ways of expressing the same concept were treated equivalently. Additionally, lemmatization was applied to reduce words to their base or root forms. This process transformed inflected words such as "mencari" (searching) into their root form "cari" (search). By focusing on the core meaning of words, lemmatization enhanced the system's ability to generalize across different forms of similar terms, improving the accuracy of both intent detection and entity recognition.

Finally, vectorization techniques such as Term Frequency-Inverse Document Frequency (TF-IDF) and word embeddings were used to convert text into numerical representations. These numerical forms allowed machine learning algorithms to process the data effectively, preserving semantic relationships and contextual meaning. The combination of these preprocessing steps ensured that the dataset was not only consistent and noise-free but also compatible with the machine learning algorithms used in the system. This meticulous preparation enhanced the performance of the models, ultimately contributing to the success of the NLP system.

3.4 Model Training

NLP models were trained using Gemini.ai, a machine learning platform that supports rapid prototyping and iterative development of machine learning systems. The primary objective of the training process was to enable the system to accurately classify user queries and extract relevant entities. This training focused on two fundamental tasks: intent detection and entity recognition. For intent detection, the system was designed to categorize user queries into predefined intent classes, such as "hotel search" or "transportation selection." This capability ensured that the system could understand the general purpose behind user inputs. Simultaneously, the entity recognition task aimed to extract specific, meaningful components from the queries, tagging them with appropriate labels such as "hotel," "bintang 5," or "bus." These tasks collectively enabled the NLP system to process and interpret user requests in a structured and actionable format.

The dataset used for training consisted of 200 annotated samples. However, it is important to note that this dataset was primarily intended for testing and evaluation purposes rather than comprehensive training. The small dataset size allowed for controlled experiments and quick iterations during development. In practical application, the system does not impose a limit on the number of queries it can process, as it relies on a more extensive and dynamically generated dataset during deployment.

To address the challenges posed by the limited dataset size, data augmentation techniques were employed to enrich the training data. This process involved generating additional variations of the existing queries by paraphrasing and introducing minor modifications, which expanded the dataset's diversity without compromising its core semantics. These augmented samples allowed the model to learn more robustly and generalize effectively across a broader range of inputs. For instance, a query like "Carikan saya hotel bintang 5 dengan bus" could be augmented to variations such as "Tolong temukan hotel bintang lima yang menggunakan transportasi bus" or "Saya butuh hotel bintang 5 dengan moda bus." Training process visualization in Google AI Studio can be seen in Figure 2.

3.5 Integrated with the system

The integration of the trained NLP models into the system's backend was a crucial step in enabling real-time query processing and delivering an efficient user experience. Upon receiving a user query, the system activates the trained models to interpret the input, beginning with the identification of the user's intent and the extraction of relevant entities from the text. For instance, in a query such as *"Carikan saya hotel bintang 5 dengan bus,"* the models identify the intent as a hotel search and extract entities like *"hotel"* as the category, *"bintang 5"* as the star rating, and *"bus"* as the transportation mode. These extracted entities are then translated into structured parameters and mapped to predefined IDs within the database schema. can be seen in Figure 3.

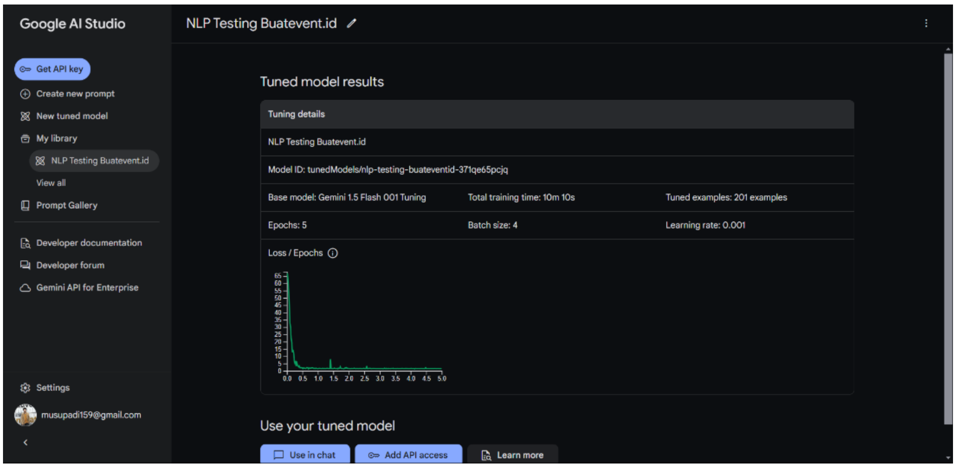

Figure 2. Training process visualization in Google AI Studio.

The integration process involved creating a seamless connection between the NLP pipeline and the system’s core functionalities. Once the intent and entities are mapped to database-compatible parameters, they are used to construct dynamic queries that retrieve relevant information. For example, the extracted parameters might prompt the system to search for five-star hotels that offer bus transportation within the specified budget constraints. This automated interaction between the NLP models and the database ensures accurate and timely results that align with user preferences.

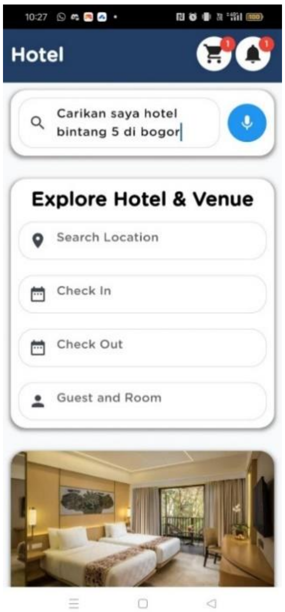

Figure 3. system integration where the user query
'Carikan saya hotel bintang 5 di Bogor' is processed by the NLP model.

To further enhance usability, the system was designed to handle a wide range of input variations and dynamically adapt to user requests. By utilizing real-time query processing, the system minimizes latency and provides immediate responses, significantly reducing the effort required for event planning. This integration not only simplifies complex query handling but also ensures scalability, allowing the system to accommodate diverse scenarios and user requirements effectively. The result is a robust, AI-powered platform that streamlines the decision-making process, optimizes resource allocation, and improves the overall user experience.

4. RESULT AND DISCUSSION

This study developed an AI-powered event planning system that integrates intent classification, entity recognition, and personalized budget optimization. The system was evaluated based on its ability to accurately interpret user queries and deliver context-appropriate responses, particularly within the domain of corporate event planning.

Model evaluation was conducted using 5-fold stratified cross-validation on the 200 annotated query samples. The intent classification module achieved an average accuracy of 92.5 percent. In addition to accuracy, further metrics were calculated to assess model robustness. The precision and recall scores for intent detection were 91.8 percent and 90.4 percent respectively, resulting in an F1-score of 91.1 percent. For entity recognition, the Conditional Random Fields (CRF) model achieved an average F1-score of 89.3 percent across key entity classes such as "hotel," "transportation," and "location."

While these results indicate high model reliability, it is important to acknowledge that the dataset was relatively small. To address potential overfitting, the model relied on transfer learning from pre-trained FastText embeddings and employed cross-validation to mitigate bias. Nonetheless, the limited scope of annotated samples may restrict the model's generalizability to highly ambiguous or complex real-world queries. Therefore, further validation using larger and more diverse real-world data is necessary.

The system was also tested in a real-world scenario during a corporate training event planning process in Jakarta. The AI-assisted interface successfully interpreted user input queries such as "Plan a two-day seminar for 80 people in Jakarta with budget-friendly transport and catering" and returned optimized results with personalized vendor suggestions within seconds. Compared to a manual planning benchmark, the system reduced the time required for vendor matching and budget adjustment by approximately 30 percent.

No baseline system was available for direct computational comparison. However, when compared qualitatively with keyword-based search and form-based tools, the AI-driven approach provided a significantly more dynamic and personalized user experience. In summary, the NLP model demonstrated strong performance in both controlled and practical use cases. However, continued improvement through model retraining, user feedback integration, and expansion of annotated datasets will be necessary to enhance the system’s adaptability to real-world planning contexts.

4.1 Chalanges

One of the main challenges encountered in this study was the limited size of the dataset. While the dataset was sufficient for initial testing, its size restricted the system's ability to handle highly varied or unconventional queries. For instance, phrases containing uncommon terminology or multilingual components presented difficulties for the model.

Another significant challenge was the complexity of real-world queries. For example, a query like *"Carikan hotel bintang lima di Bali dengan layanan bus"* required precise extraction of entities and dynamic adjustments to query parameters. This level of complexity emphasized the importance of continuous fine-tuning and enriching the training dataset to enhance the model's ability to generalize.

Furthermore, integrating the system with real-time backend processing posed difficulties in maintaining low latency, particularly when managing simultaneous queries from multiple users. Ensuring scalability and consistent performance under these conditions demanded substantial optimization of the backend architecture.

4.2 Discussion of Limitation

The limited dataset size was a significant limitation, as the system relied on only 200 samples for training and testing due to platform constraints. While data augmentation techniques partially mitigated this limitation, a larger and more diverse dataset would enhance the system's robustness and generalization capabilities. This limitation also affected the system's ability to handle domain-specific queries involving rare or highly specific terms.

Hardware constraints further impacted the training process. While the system was trained on Gemini.ai’s platform, which supports rapid prototyping, the computational resources available limited the scope of the training. The training process required approximately **2 hours**, reflecting the computational demand of fine-tuning NLP models. The integration of the NLP model into a real-time application also presented challenges in terms of compatibility and scalability. Certain queries required additional preprocessing or adjustments to align with database constraints, occasionally causing minor delays in response time.

Despite these limitations, the system performed exceptionally well in most scenarios, demonstrating its potential as an efficient tool for corporate event planning. The ability to handle diverse queries and provide actionable recommendations highlights the effectiveness of the underlying AI technologies.

5. CONCLUSION AND FUTURE WORK

5.1 Conclusion

This study successfully demonstrated the integration of artificial intelligence into corporate event planning, specifically through natural language processing and budget optimization. The system developed in this research, powered by Gemini.ai and supported by FastText-based embeddings, achieved 92.5 percent accuracy in intent classification and over 89 percent F1-score in entity recognition. These results were obtained from 5-fold cross-validation on a manually annotated dataset of 200 queries.

Beyond technical performance, the platform was evaluated in a real-world use case involving a corporate training event in Jakarta. Compared to manual planning methods and static form-based tools, the AI-driven system reduced planning time by approximately 30 percent, demonstrating improved efficiency and usability. This highlights the platform's practical value in streamlining decision-making, reducing operational delays, and offering cost-effective solutions.

However, some limitations remain. The current version was trained and tested on a relatively small dataset, raising potential concerns about overfitting and generalization. In addition, although the system performed well in a controlled deployment, larger-scale implementation across multiple concurrent users will require further backend optimization to ensure scalability and low-latency performance. Feedback from early users was collected through post-interaction surveys and integrated into system refinement cycles. This feedback informed improvements in intent detection accuracy, UI responsiveness, and the clarity of vendor recommendations. Continuous integration of user feedback will remain central to system improvement and user-centered design.

5.2 Future Work

To enhance model robustness, future development will focus on expanding the annotated dataset using real-world queries sourced from platform usage. Incorporating domain-specific terms and rare expressions will allow the system to better handle complex or ambiguous requests. Technically, future iterations will explore the use of advanced transformer-based architectures such as BERT or IndoBERT for improved performance in intent detection and entity recognition. These models are expected to improve contextual understanding, especially in long or multi-intent queries. From a deployment perspective, optimizing system scalability is also a priority. This includes adapting backend infrastructure to support concurrent users, minimizing response times, and improving data caching mechanisms.

Finally, multilingual support will be explored by integrating translation modules and training on datasets in multiple languages. This enhancement will expand the system’s usability for international clients and events, reinforcing its potential as a global event planning solution.

REFERENCES

[1] Monfort, A., Villagra, N., & Sánchez, J. (2021). "Economic impact of corporate foundations: An event analysis approach." *Journal of Business Research, 122,* 159-170 DOI: 10.1016/j.jbusres.2020.08.046

[2] Bahri, S., & Ramaditya, M. (2024). "Innovation of the finished banknotes business model changes in Indonesian banknotes printing." *Journal of Open Innovation: Technology, Market, and Complexity, 10(3)*, 100323. DOI:

[3] Han, X., Meng, X., & Zhang, Y. (2025). "Exploiting multiple influence pattern of event organizer for event recommendation." *Information Processing & Management, 62(2),* 103966. DOI: 10.1016/j.ipm.2024.103966

[4] Wünderlich, N. V., Hogreve, J., Chowdhury, I. N., Fleischer, H., Mousavi, S., Rötzmeier-Keuper, J., & Sousa, R. (2019). "Overcoming vulnerability: Channel design strategies to alleviate vulnerability perceptions in customer journeys." *Journal of Business Research*. DOI: 10.1016/j.jbusres.2019.07.027

[5] Wu, F., Sorokina, N., & Putra, E. D. (2023). "Customers Satisfaction on Robots, Artificial Intelligence and Service Automation (RAISA) in the Hotel Industry: A Comprehensive Review." *Open Journal of Business and Management*, 11(3). DOI: 10.4236/ojbm.2023.113069

[6] Silva, F. J. F., Câmara, G. F. M., Vieira, J. A. C., & Santos, C. A. S. M. (2020). "Is the spending behaviour of tourists affected by low-cost carriers' operation? Some empirical evidence." *Tourism Management Perspectives*, 33, 100630. DOI: 10.1016/j.tmp.2019.100630

[7] Flutter Documentation. Flutter.dev, https://flutter.dev/docs.

[8] The Go Programming Language. Golang.org, https://golang.org/doc/.

[9] PHP: Hypertext Preprocessor. PHP.net, https://www.php.net/docs.php

[10] Python Documentation. Python.org, https://docs.python.org/.

[11] MySQL Reference Manual. MySQL.com, https://dev.mysql.com/doc/.

[12] Ahmed, N., Saha, A. K., Noman, M. A. A., Jim, J. R., Mridha, M. F., & Kabir, M. M. (2024). *Deep learning-based natural language processing in human–agent interaction: Applications, advancements and challenges.* Natural Language Processing Journal, 9, 100112. DOI: 10.1016/j.nlp.2024.100112

[13] Hardiman, J. P. W., Thio, D. C., Zakiyyah, A. Y., & Meiliana. (2024). *AI-powered dialogues and quests generation in role-playing games using Google's Gemini and Sentence BERT framework.* Procedia Computer Science, 245, 1111–1119. DOI: 10.1016/j.procs.2024.10.340

[14] Darraz, N., Karabila, I., El-Ansari, A., Alami, N., & El Mallahi, M. (2025). *Integrated sentiment analysis with BERT for enhanced hybrid recommendation systems.* Expert Systems with Applications, 261, 125533. DOI: 10.1016/j.eswa.2024.125533

[15] Jang, H. (2019). *A decision support framework for robust R&D budget allocation using machine learning and optimization.* Decision Support Systems, 121, 1–12. DOI: 10.1016/j.dss.2019.03.010

[16] Zhou, J., Chi, Y., Wen, Y., & Gao, K. (2024). TourismNER: A Tourism Named Entity Recognition method based on entity boundary joint prediction. Expert Systems with Applications. DOI: 10.1016/j.iswa.2024.200447

[17] Xu, Y., Meng, J., Zhang, Y., & Koo, J. (2024). *Present bias, mental budget constraint, and the payday consumption cycle.* China Economic Review, In Press, Journal Pre-proof, 102306. DOI: 10.1016/j.chieco.2024.102306

[18] Zheng, W., Zaman, A., Farooq, F., Althoey, F., Alaskar, A., & Akbar, A. (2023). *Sustainable predictive model of concrete utilizing waste ingredient: Individual algorithms with optimized ensemble approaches.* Materials Communications, 105901. DOI: 10.1016/j.mtcomm.2023.105901

[19] Mahapatra, B. S., Ghosh, D., Pamucar, D., & Mahapatra, G. S. (2025). *Dynamic group decision-making for enterprise resource planning selection using two-tuples Pythagorean fuzzy MOORA approach.* Expert Systems with Applications, 263, 125675. DOI: 10.1016/j.eswa.2024.125675

[20] Song, X., Fu, X., Ren, M., Pace, P., & Aloi, G. (2024). *Prediction-based data collection of UAV-assisted Maritime Internet of Things.* Vehicular Communications, 50, 100854. DOI: 10.1016/j.vehcom.2024.100854

[21] Islam, M. R., Ahmad, A., & Rahman, M. S. (2024). *Bangla text normalization for text-to-speech synthesizer using machine learning algorithms.* Journal of King Saud University - Computer and Information Sciences, 36(1), 101807. DOI: 10.1016/j.jksuci.2023.101807

[22] Quarto, M., D'Urso, G., & Giardini, C. (2022). *Micro-EDM optimization through particle swarm algorithm and artificial neural network.* Precision Engineering, 73, 63–70. DOI: 10.1016/j.precisioneng.2021.08.018

[23] Dolha, D. N., & Buchmann, R. A. (2024). Experiments with natural language queries on RDF vs. XML-serialized BPMN diagrams. Procedia Computer Science, 246, 3246–3255. DOI: 10.1016/j.procs.2024.09.315