DEVELOPING AN LLM-BASED CHATBOT FOR UNIVERSITY ADMISSION COUNSELING

Pham Ngoc Tinh[1,2,*],
Nguyen Nang Hung Van[1], Ngo Truong An[2]

DOI: https://doi.org/10.57001/huih5804.2025.422

ABSTRACT

In the context of digital transformation in education, this study presents the development of an intelligent chatbot system for university admissions counseling, utilizing Retrieval-Augmented Generation (RAG) architecture and Large Language Models (LLMs). The system is built upon a specialized dataset collected from official sources at Dong A University. The study conducts detailed experimental evaluations, comparing the performance of various embedding models and retrieval methods, including hybrid search. Furthermore, the paper analyzes the response quality of several prominent LLMs using standard semantic and syntactic evaluation metrics. Experimental results show that the proposed RAG architecture, when combined with an appropriate LLM, achieves superior performance and fast response times. This confirms the system's effectiveness in providing accurate admissions information, reducing the workload for counseling staff, and enhancing user experience.

Keywords: *Natural Language Processing (NLP), Chatbot, Artificial Intelligence in Education, Large Language Model (LLM), Retrieval-Augmented Generation (RAG).*

[1]University of Danang - University of Science and Technology, Vietnam
[2]Faculty of Information Technology, Dong A University, Vietnam
[*]Email: tinhpn@donga.edu.vn

Received: 28/8/2025
Revised: 14/10/2025
Accepted: 28/11/2025

1. INTRODUCTION

The rapid development of Artificial Intelligence (AI) is reshaping many domains of modern life. Among these, the emergence of Large Language Models (LLMs) [1] built on Natural Language Processing (NLP) techniques [2] has enabled intelligent communication systems that interact with humans more flexibly and naturally. NLP acts as a bridge between natural and machine language, allowing computers to understand, process, and respond not only with semantic accuracy but also in a contextually appropriate manner [3]. LLMs have therefore become useful tools across domains such as education [4], economics [5], and beyond; yet their wide adoption also surfaces challenges, from ensuring accuracy and reliability to adapting to domain-specific contexts, which call for careful design and optimization in each application.

In educational counseling and learning support, combining LLMs with Retrieval-Augmented Generation (RAG) has shown promise, as institution-specific evidence can ground model outputs and improve user trust when tailored to concrete educational goals [6]. Even so, deployments must still address response accuracy, privacy and ethics around user data, and quality-latency-cost trade-offs.

This study develops an intelligent chatbot for Vietnamese university admissions counseling using Dong A University resources. The knowledge base includes frequently asked questions, official university documents, and interaction logs from actual counseling sessions, enabling the system to handle diverse queries with timely and accurate information. Built on modern NLP techniques, the system combines embedding-based retrieval with a carefully curated and pre-processed institutional corpus. Initial evaluations suggest the system can reduce advisor workload and improve information accessibility and user experience. Future extensions include multilingual support, integration with social platforms, and personalized guidance based on candidate profiles.

2. BACKGROUND AND RELATED WORK

University admission counseling is critical for guiding prospective students, yet human-only workflows struggle to scale to large, simultaneous query volumes. As demand for real-time support increases, AI particularly LLM-based assistants has been explored in higher education, with benefits reported when answers are constrained to domain sources and governed by appropriate safeguards [4]. However, in high-accuracy scenarios such as admissions, standalone LLMs face well-documented limitations: hallucination (plausible but incorrect content) and stale knowledge due to fixed training cut-offs, while policies and quotas update annually making unconstrained deployment risky for candidates' decisions [6].

To mitigate these risks, Retrieval-Augmented Generation (RAG) is widely regarded as an effective approach [7]. Rather than relying solely on parametric memory, the system retrieves relevant passages from a trusted, institution-specific corpus (e.g., official admissions texts) and conditions the generation step on that evidence, which helps maintain factuality and recency in counseling contexts. Prior studies in education and counseling report that pairing LLMs with retrieval over institutional documents can improve answer reliability and user acceptance, and that quality should be evaluated with standard text metrics together with practical indicators such as usability and latency [8].

Positioned against this literature, our work focuses on a complete, context-adapted RAG implementation for Dong A University and an empirical analysis of core components embeddings, keyword/dense/hybrid retrieval, and modern LLMs under evaluation settings aligned with real advising workflows. This deployment-oriented perspective complements prior studies by emphasizing document grounding, citation-first responses, and measurement of both linguistic quality and responsiveness.

3. MATERIALS AND METHODS

3.1. Problem Formulation

Mathematically, we define the university admission counseling task as an optimization problem, where the system's objective is to find the most suitable answer a_q for a user's query q. Let:

$\mathcal{Q}$ is the collection of all user input questions (e.g., 'How long is the Nursing program?').

$\mathcal{D} = \{d_1, d_2, \ldots, d_N\}$ is a set of N admissions-related documents, where each d_i is a text string.

Embedding fembed: $T \rightarrow \mathbb{R}^k$ and $T = \mathcal{D} \cup \mathcal{Q}$ natural-language text to a k-dimensional vector (e.g., Alibaba-NLP/gte-multilingual-base with k = 768).

$\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ is the set of representation vectors for the documents, where v_i = fembed(d_i) and $d_i \in \mathcal{D}$

v_q= fembed(q) is the representation (embedding) vector of the query q ∈ $\mathcal{Q}$.

The RAG system's workflow consists of two main stages: **Retrieval** and **Generation**.

Retrieval: We define a similarity function $S : \mathbb{R}^k \times \mathbb{R}^k \rightarrow [-1, 1]$, based on cosine similarity to quantify the relevance between a query and a document:

$$S_{cosine}(v_{q,} v_i) = \frac{v_q . v_i}{\|v_q\| . \|v_i\|} \tag{1}$$

The retrieved set R_q comprises the K documents with the highest similarity to the query:

$$R_q = \{ d_{\mathrm{i}} \in \mathcal{D} | v_i \ is \ in \ top - K \ by \ S_{cosine}(v_{q,} v_i)\} \tag{2}$$

Equation (2) selects the $top - K$ documents with the highest similarity to the query, ensuring the chatbot bases its response on the most informative evidence.

Hybrid Retrieval Scoring: We fuse normalized semantic and lexical signals into a single hybrid score and select $top - K$ passages [9].

$$S_{vector}(v_{q,} v_i) = \frac{v_q . v_i}{\|v_q\| . \|v_i\|} \in [-1, 1] \tag{3}$$

$$S_{keyword}(q, \ d_i) = \frac{|Tok(q) \cap Tok(q)|}{|Tok(q) \cup Tok(q)|} \in [0, 1] \tag{4}$$

Where $Tok(.)$ denotes the tokenization function and S_{vector} maps cosine similarity from [– 1, 1] to [0 , 1]. The mixing weight $\alpha \in [\ 0\ ,\ 1\ ]$ is tuned on a labeled development.

Hybrid score:

$$S_{hybrid}(q, \ d_i) = \alpha . S_{vector}(v_{q,} v_i) + (1 - \alpha) . S_{keyword}(q, \ d_i) \tag{5}$$

The updated retrieval rule is:

$$R_q = \{ d_{\mathrm{i}} \in \mathcal{D} | v_i \ is \ in \ top - K \ by \ S_{hybrid}(q, \ d_i)\} \tag{6}$$

Prompt Construction: Given a user query q and its retrieved set $R_q \subseteq D$, the system builds a textual input sequence for the LLM. We define a context–construction function:

$$P : \mathcal{Q} \times 2^{\mathcal{D}} \rightarrow S(v_q, v_i)$$

where $2^{\mathcal{D}}$ is the power set of the document collection D, and $S(v_q, v_i)$ denotes the similarity measure between the query embedding v_q and a document embedding v_i.

The instantiated context string is constructed as:

$$P(q, \mathcal{R}_q) = \text{Query:"+q+Context"} + concatenate(\mathcal{R}_q) \tag{7}$$

Equation (7) constructs the prompt by concatenating the query with the retrieved documents, preparing the LLM context.

Generation: A large language model, $F_{LLM} : S \to \mathcal{A}$, generates the final answer $\mathrm{a}_{\mathrm{q}}^{*}$ by synthesizing the information from the retrieved context R_q and the original query q.

$$\tilde{a}_q = F_{LLM}\left(P\left(q, \mathcal{R}_q\right)\right)$$

Optimization: We aim to maximize the expected agreement between the model output and a reference answer $a_q^* \in \mathcal{A}$:

$$\max_{\mathcal{R}_q,P} E_{q\sim Q}[\text{Accuracy}(\tilde{a}_q, a_q^*)] \qquad (8)$$

In which $\text{Accuracy}(\tilde{a}_q, a_q^*)$ is an evaluation functional comparing the generated answer with the gold reference; MRR@k for retrieval-focused objectives. The expectation is taken over the query distribution, i.e., results are averaged across all queries.

Equation (8) specifies the objective: maximize the accuracy of generated answers with respect to the reference (ground-truth) responses.

3.2. System Architecture

The system operates through a modular, four-step process as depicted in Figure 1, detailed as follows:

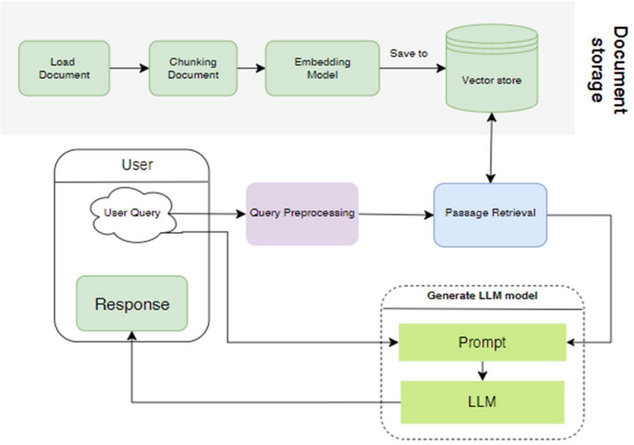


Figure 1. Query processing workflow in a chatbot

Step 1: User Interface

Users access the chatbot via a web interface, enter questions in Vietnamese, and receive detailed responses related to the admissions process.

Example (user query): "GIỚI THIỆU NGÀNH ngành công nghệ thông tin (CNTT)?"

Expected response: "Công nghệ thông tin là một lĩnh vực sử dụng hệ thống máy tính, phần mềm và mạng để thu thập, lưu trữ, xử lý và truyền tải thông tin. Đây là ngành then chốt trong phát triển kinh tế - xã hội, ứng dụng rộng ở kinh doanh, giáo dục, y tế, giải trí,..."

Step 2: Query Preprocessing

User questions are preprocessed via tokenization and stopword removal to improve retrieval accuracy and produce responses tailored to the admissions context.

Tokenization: split the query into tokens:

- Input : GIỚI THIỆU NGÀNH ngành công nghệ thông tin (CNTT)?

- Tokens: ["GIỚI", "THIỆU", "NGÀNH", "ngành", "công", "nghệ", "thông", "tin", "(CNTT)", "?"]

Tokens after removal: ["giới", "thiệu", "công", "nghệ", "thông", "tin", "cntt"]

Step 3: Passage Retrieval

After preprocessing, we embed the query, compute cosine similarity against passage embeddings, and select the top-matching passages as context for generation.

Cosine similarity formula:

$$Cosine\ Similarity(A, B) = \frac{A\,.\,B}{\parallel A \parallel \parallel B \parallel}$$

Where A is the query vector and B is the vector of a passage in the dataset [10].

Step 4: Contextual Understanding and Response Generation

The retrieved passages and the user query are combined into a single prompt and fed to the eGPT language model. The model analyzes the content and generates a coherent, context-appropriate response that aims to be factually accurate.

Example:

Query: "Em sẽ nhận được những loại bằng gì sau khi ra trường?"

Retrieved evidence: "Sau khi tốt nghiệp ra trường, em sẽ được nhận các bằng cấp (thuộc hệ thống văn bằng quốc gia) như sau..."

Generated response: "Sau khi tốt nghiệp, em sẽ nhận được bằng Cử nhân hoặc Kỹ sư theo hệ thống văn bằng quốc gia. Ngoài ra, …"

4. EXPERIMENTS AND RESULTS

4.1. Experimental Setup

Dataset: The dataset was constructed for Vietnamese admissions counseling at Dong A University (UDA). Sources include the Admissions Office (real candidate questions and approved internal materials) and public information on the university's admissions website/portal (policies, yearly admissions booklets, official announcements). All documents are UTF-8 normalized, de-duplicated, standardized for common abbreviations (e.g., "CNTT"), and retain metadata (URL, title, effective date, section anchors) to support provenance/citations in answers.

The experimental snapshot comprises over 800 data points spanning the primary admissions topics (Table 1).

Table 1. Distribution of questions in the experimental snapshot by topic

Topic	n	% of total
Programs / Majors	331	40.9
Academic Programs / Curriculum	136	16.8
Admissions	102	12.6
Tuition	98	12.1
Career Opportunities	78	9.6
Student Support & Activities	37	4.6
Facilities	17	2.1
Scholarships	13	1.6
Total	**810**	**100**

Evaluation Metrics: We assess the effectiveness and overall performance of the admissions-counseling chatbot using the following metrics and report corpus-level scores.

1. **Retrieval**: Precision@k (P@k), Recall@k (R@k), and MRR@k against a gold evidence set measure whether the system surfaces the correct passages into the context window.

2. **Generation**: Accuracy is computed against expert gold answers when available; we also report BLEU-4 and ROUGE-1/2/L for text quality [11, 12]. BLEU uses standard smoothing; ROUGE follows toolkit defaults. In addition, we track end-to-end latency to reflect real-time counseling constraints.

3. **Semantic similarity**: Cosine similarity between answer and reference embeddings is reported as a supporting indicator of semantic closeness.

4.2. Results and Analysis

4.2.1. Embedding Model Selection

Table 2. Comparative Analysis and Performance of Embedding Models

Model Name	BAAI/ bge-m3	Alibaba-NLP/gte-multilingual-base	all-MiniLM-L6-v2	text-embedding-3-small
Top 1 Accuracy	**0.7**	0.477	0.66	0.683
Top 3 Accuracy	0.843	0.72	0.82	**0.853**
Top 5 Accuracy	0.887	0.803	0.85	**0.903**
Top 10 Accuracy	0.94	0.897	0.897	**0.94**
Context Length	8192	512	256	8.192
Embedding Dimension	1024	768	384	1536
Average Embedding Time per Question (s)	0.19	0.1	**0.04**	0.5
Average Embedding Time per Sentence (s)	0.93	0.4	0.05	**0.3**

The embedding backbone critically shapes a chatbot's retrieval accuracy, latency, and compute footprint. Comparing candidates side-by-side clarifies trade-offs and helps align the final choice with real-time, domain-specific constraints. In this work, we benchmark five encoders for semantic retrieval: BAAI/bge-m3 [13], Alibaba-NLP/gte-multilingual-base [14], all-MiniLM-L6-v2 [15], and text-embedding-3-small [16]. Evaluate them using Top-K accuracy, average embedding time and related efficiency measures (see Table 2).

We designate *Alibaba-NLP/gte-multilingual-base* as our primary embedding model. In practice, it offers dependable retrieval quality, a balanced vector size, low embedding latency, and modest resource demands properties that suit real-time deployment on constrained hardware.

Table 2 summarizes our comparison of open models in a RAG setting. We report Top-K Accuracy (the share of correct answers appearing within the top-ranked results, e.g., top-1/3/5/10) to gauge ranking effectiveness for admissions queries.

We select Alibaba-NLP/gte-multilingual-base as the primary embedding model because it: (i) consistently retrieves relevant evidence under our hybrid + top-K policy; (ii) uses a moderate dimension for a compact, memory-efficient index; (iii) encodes with low latency for interactive use and (iv) supports Vietnamese/English well.

4.2.2. Performing Retrieval Methods

After embedding the corpus into a vector store, we benchmark three retrieval strategies prior to LLM integration to verify evidence quality: *Vector retrieval, Keyword retrieval, Hybrid retrieval*. To evaluate the effectiveness of our retrieval methods, we employ Precision@3 (P@3), Recall@3, and Mean Reciprocal Rank@3 (MRR@3) [9].

Table 3. Performance Comparison of Retrieval Methods

Model	P@3	R@3	MRR@3
KEYWORD	0.4333	0.6867	0.5328
VECTOR	0.3833	0.7200	0.5844
HYBRID	**0.5022**	**0.7533**	**0.6661**

As reported in Table 3, the hybrid retriever exceeds both vector and keyword baselines on all metrics, indicating that combining semantic embeddings with lexical overlap yields stronger ranking. We therefore integrate the hybrid configuration to optimize retrieval performance.

4.2.3. Evaluating Generated Answer Quality

The performance of three large language models eLLaMA, eGPT, and eDEEPSEEK are summarized in the following tables based on our experimental results. The evaluation metrics include BLEU, ROUGE, BERTScore, and execution time, enabling a comparative assessment of each model's overall effectiveness in generating natural-language responses on the test dataset.

Table 4. Comparison of BLEU and ROUGE scores across models

Model	BLEU	ROUGE-1	ROUGE-2	ROUGE-L
eGPT	0.3691	0.6471	0.5314	0.5191
eDEEPSEEK	0.2400	0.6071	0.4579	0.4769
eLLAMA	0.1312	0.3418	0.2400	0.2650

Table 4 reports BLEU and ROUGE for each model. eGPT attains the highest lexical scores (BLEU = 0.3691; ROUGE-1 = 0.6471; ROUGE-2 = 0.5314; ROUGE-L = 0.5191), indicating stronger n-gram overlap, summary alignment with references. eDEEPSEEK ranks second on all four metrics , while LLaMA shows the lowest. These results suggest that, on our corpus, eGPT offers the most consistent lexical precision and coverage among the three LLMs.

Table 5. Comparison of execution time and BERTScore across models

Model	BERTScore-P	BERTScore-R	BERTScore-F1	Average generation time (s)
eGPT	0.7603	0.8040	0.7800	4178.2
eDEEPSEEK	0.7422	0.7611	0.7500	14217.3
LLaMA	0.6908	0.6624	0.6720	17337.4

Table 5 compares semantic similarity (BERTScore) and generation time. eGPT again leads with the highest BERTScore-F1 and the fastest average generation time. eDEEPSEEK yields a lower BERTScore-F1 with substantially longer, and LLaMA records the lowest BERTScore-F1 with comparable latency. The comparison underscores a typical quality–latency trade-off, yet in this snapshot eGPT provides both stronger semantic alignment and lower delay, making it the most favorable choice for real-time admissions counseling.

Overall, eGPT gives the best real-time trade-off (strong lexical/semantic scores, lowest latency). eDEEPSEEK is slower but semantically strong when quality matters more than speed; LLaMA trails on accuracy/BERTScore. All beat rule-based, but still below human advisors.

4.2.4. Analysis of BLEU Metrics Across Models

Figure 2 (BLEU heatmap) compares eGPT, eDEEPSEEK, and LLaMA across n-gram precisions (P-1…P-4), brevity penalty (BP), and length ratio (LR). eDEEPSEEK achieves the strongest unigram precision (P-1 = 0.5504, better keyword coverage), while eGPT leads on higher n-grams (P-3 = 0.3281; P-4 = 0.2955, stronger phrase coherence); LLaMA is consistently lower. Brevity/length further differentiate them: eDEEPSEEK has the mildest penalty and closest length to references (BP = 0.6441; LR = 0.6945), eGPT is moderate (BP=0.5968; LR = 0.6596), and LLaMA is markedly short (BP = 0.1335; LR = 0.331), indicating under-generation and potential information loss.

Overall, the heatmap indicates a trade-off between lexical coverage and multi-token fluency: eDEEPSEEK favors coverage, eGPT better sustains coherent multi-phrase responses, and LLaMA requires tuning to mitigate brevity underscoring the need to balance precision and length when optimizing counseling outputs.

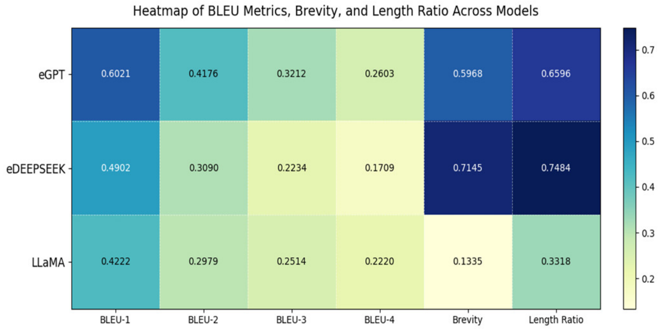
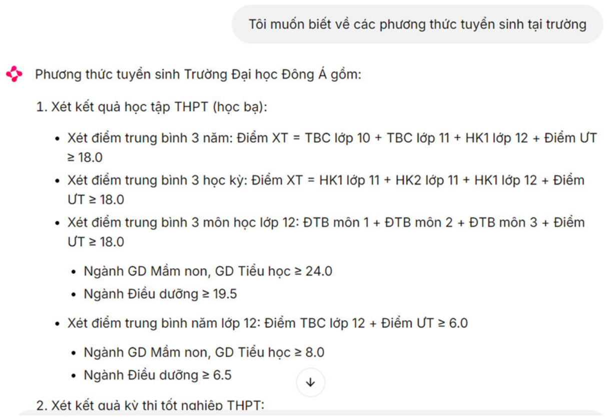


Figure 2. Heatmap of BLEU metrics across models

4.3. Developing the User Interface with the Chainlit Framework

We build the admissions-chatbot UI with Chainlit, an open-source Python framework for LLM apps, which provides ready-made components and flexible customization to accelerate development while keeping the interface intuitive [17]. This lets us focus on dialogue flows and personalization; the deployed UI is responsive across devices and integrates tools for efficient admissions information retrieval (see Figure 3).

Additionally, according to Zhang and Lee [18], Chainlit effectively supports rapid prototyping of chatbot variants through direct integration with language-model APIs, thereby accelerating system evaluation and improvement.



Tôi muốn biết về các phương thức tuyển sinh tại trường

Phương thức tuyển sinh Trường Đại học Đông Á gồm:

1. Xét kết quả học tập THPT (học bạ):
   - Xét điểm trung bình 3 năm: Điểm XT = TBC lớp 10 + TBC lớp 11 + HK1 lớp 12 + Điểm ƯT ≥ 18.0
   - Xét điểm trung bình 3 học kỳ: Điểm XT = HK1 lớp 11 + HK2 lớp 11 + HK1 lớp 12 + Điểm ƯT ≥ 18.0
   - Xét điểm trung bình 3 môn học lớp 12: ĐTB môn 1 + ĐTB môn 2 + ĐTB môn 3 + Điểm ƯT ≥ 18.0
     - Ngành GD Mầm non, GD Tiểu học ≥ 24.0
     - Ngành Điều dưỡng ≥ 19.5
   - Xét điểm trung bình năm lớp 12: Điểm TBC lớp 12 + Điểm ƯT ≥ 6.0
     - Ngành GD Mầm non, GD Tiểu học ≥ 8.0
     - Ngành Điều dưỡng ≥ 6.5
2. Xét kết quả kỳ thi tốt nghiệp THPT:

Figure 3. Chatbot interface example

5. CONCLUSION

This paper presented the design and deployment of a RAG-based admissions-counseling chatbot built on large language models to provide accurate, timely information to prospective students. In deployment, the system handled Vietnamese effectively, generated context-appropriate responses, improved user experience during information seeking and application support.

The system is anchored in Dong A University's admissions corpus, ensuring domain fidelity but limiting breadth. Two limitations remain: (i) dependence on institution-specific data that evolves across cycles, (ii) sensitivity to very short or abbreviation-heavy queries beyond the curated scope. These motivate continued retrieval refinement and routine corpus updates.

Future work will: (i) extend multilingual support (Vietnamese/English) and add speech I/O for voice interactions; (ii) incorporate personalization from prior user behavior; and (iii) exploit graph-structured knowledge to strengthen context-aware retrieval and reduce errors enhancing reliability, responsiveness, and scalability for broader educational counseling use.

REFERENCES

[1]. L. Qin, Q. Chen, X. Feng, Y. Wu, Y. Zhang, Y. Li, M. Li, W. Che, P. S. Yu, "Large Language Models Meet NLP: A Survey," *arXiv(Publication_Name), vol. abs/2405.12819*, 2024.

[2]. D. Khurana, A. Koli, K. Khatter, S. Singh, "Natural Language Processing: State of the Art, Current Trends and Challenges," *Multimedia Tools and Applications*, 82, 3, 3713-3744, 2023.

[3]. T. Beysolow II, "What Is Natural Language Processing?," in *Applied Natural Language Processing with Python: Implementing Machine Learning and Deep Learning Algorithms for Natural Language Processing*, Berkeley, CA: Apress, 2018.

[4]. M. Abedi, I. Alshybani, M.R.B. Shahadat, M. Murillo, "Beyond Traditional Teaching: The Potential of Large Language Models and Chatbots in Graduate Engineering Education," *Qeios*, 2023.

[5]. M. Steve, S. Brightwood, O. Godwin, *Implementing AI Chatbots for Real-Time Supply Chain Monitoring and Risk Management*. 2024.

[6]. B. Memarian, T. Doleck, "ChatGPT in education: Methods, potentials, and limitations," *Computers in Human Behavior: Artificial Humans*, 1, Art. 100022, 2023.

[7]. P. Lewis, et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," *NeurIPS*, 2020.

[8]. N. N. H. Van, P. H. Do, V. N. Hoang, T. T. K. Nguyen, M. T. Pham, "AI-Powered University Admission Counseling: A Use Case of Large Language Models in Student Guidance," *IEEE Transactions on Learning Technologies*, 18, 856-868, 2025.

[9]. Nguyen T. T. K., Hoang V. N., Tinh P. N., Van N. N. H., "Efficient Chatbot for university admission consultation using large language models," *The University of Danang - Journal of Science and Technology*, 23, 9A, 80-85, 2025. doi: 10.31130/ud-jst.2025.23(9A).329E.

[10]. D. M. Christopher, R. Prabhakar, S. Hinrich, *Introduction to information retrieval*. Cambridge University Press, 2008.

[11]. Papineni K., Roukos S., Ward T., Zhu W. J., "BLEU: A Method for Automatic Evaluation of Machine Translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, 311-318, 2002.

[12]. Lin C. Y., "ROUGE: A Package for Automatic Evaluation of Summaries," in *Proceedings of the Workshop on Text Summarization Branches Out* (WAS 2004), 74-81, 2004.

[13]. J. Chen, S. Xiao, P. Zhang, K. Luo, D. Lian, Z. Liu, "BGE-M3: A Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embedding Model with State-of-the-Art Performance," *arXiv preprint, Beijing Academy of Artificial Intelligence (BAAI)*, 1-12, 2024.

[14]. Kexin Pei, Wenhui Wu, Yichong Xu, Yelong Shen, Ming Gong, Jianfeng Gao, Nan Duan, "mGTE: Generalized Long-Context Text Representation and Reranking Models for Multilingual Text Retrieval," In *Proceeding(s) of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024. https://huggingface.co/Alibaba-NLP/gte-multilingual-base

[15]. N. Reimers, I. Gurevych, "Making Monolingual Sentence Embeddings Multilingual Using Knowledge Distillation," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing* (EMNLP), Association for Computational Linguistics, 4512-4525, 2020.

[16]. OpenAI, *Text Embedding-3 Small: Efficient and Scalable Text Representations*. OpenAI Technical Report, OpenAI, San Francisco, USA, Jan. 2024.

[17]. Chainlit Contributors, *Chainlit: A framework for building LLM-powered chat applications*. Chainlit Documentation, Chainlit, 2025. Accessed: 01/02/2025. [Online]. https://docs.chainlit.io/get-started/overview

[18]. H. Zhang, M. Lee, "Rapid Prototyping of Conversational AI Systems using Chainlit," *International Journal of Intelligent Systems and Applications*, AICIT, 15, 1, 45-53, 2025.