# From Questions to Insightful Answers: Building an Informed Chatbot for University Resources

Subash Neupane*, Elias Hossain*, Jason Keith†, Himanshu Tripathi*,
Farbod Ghiasi*, Noorbakhsh Amiri Golilarz*Amin Amirlatifi‡, Sudip Mittal*, Shahram Rahimi*
* *Department of Computer Science & Engineering*
*Mississippi State University*
{sn922, mh3511, ht557, fg289}@msstate.edu,{amiri, mittal, rahimi}@cse.msstate.edu
‡ *Dave C. Swalm School of Chemical Engineering*
*Mississippi State University*
{amin}@che.msstate.edu
† *Office of the Senior Vice President and Provost*
*Iowa State University*
{jkeith}@iastate.edu

***Abstract*—This research-to-practice full paper presents BARKPLUG V.2, a Large Language Model (LLM)-based chatbot system built using Retrieval Augmented Generation (RAG) pipelines to enhance the user experience and access to information within academic settings. The objective of BARKPLUG V.2 is to provide information to users about various campus resources, including academic departments, programs, campus facilities, and student resources at a university setting in an interactive fashion. Our system leverages university data as an external data corpus and ingests it into our RAG pipelines for domain-specific question-answering tasks. We evaluate the effectiveness of our system in generating accurate and pertinent responses for Mississippi State University, as a case study, using quantitative measures, employing frameworks such as Retrieval Augmented Generation Assessment (RAGAS). Furthermore, we evaluate the usability of this system via subjective satisfaction surveys using the System Usability Scale (SUS). Our system demonstrates impressive quantitative performance, with a mean RAGAS score of 0.96, and satisfactory user experience, as validated by usability assessments.**

***Index Terms*—Chatbot, LLM, RAG, University resources, information access**

## I. INTRODUCTION

Colleges and universities invest significant time and resources into enhancing their websites to effectively communicate crucial information about the institution and available campus resources. The institutional website serves as its "virtual face", the face it has chosen to present to the online world, including potential and current students, faculty, parents, alumni and general users [1]. Although these websites offer comprehensive information, they lack the capability to provide personalized responses to user queries. For instance, when a prospective student needs details about submitting ACT scores, wants to know their tuition and fees, or is unsure which parent's information to use on their FAFSA application, they must navigate through multiple webpages to find answers. This process frequently requires a considerable amount of time. Yet, at times, users' queries are left unanswered due to either unclear information or lack of personal interaction.

Various campus resources and services, such as academic departments, career centers, admissions, registration, scholarships, and financial aid, are available to assist students with both academic and non-academic queries. These resources are equipped with dedicated officers who provide guidance to students. However, they are constrained by service-time limitation (may only be available during specific working hours) and may require an appointment, which might not always accommodate busy student schedules. Additionally, delays in responses and longer wait times, particularly during application periods, can diminish prospective students' interest in the institution, ultimately affecting university revenue.

To address these challenges, universities are currently employing conversational agents, also known as chatbots, to offer support to users. Chatbots are *"software systems that mimic interactions with real people"* [2] by engaging in conversation through natural language using machine learning technology, specifically Natural Language Processing (NLP) [3]. For instance, Arizona State University (ASU) developed a chatbot named *Sunny* [4] with the intentional design to offer emotional support to students, alongside providing information regarding ASU. Sunny efficiently addresses frequently asked questions such as inquiries about financial aid, academic advisors, and accessing ASU email accounts. Similarly, Georgia State University (GSU) introduced a virtual assistant named *Pounce* [5] to tackle obstacles to enrollment faced by students transitioning from high school to college. These obstacles encompassed tasks like financial aid applications, document submissions, immunization records, placement exams, and class registration. GSU reported a notable 22% reduction in summer melt due to Pounce's assistance. Beyond admissions and enrollment, universities are increasingly deploying chatbots to aid students in their academic pursuits and campus life. One notable example is the chatbot *Beacon* [6] developed

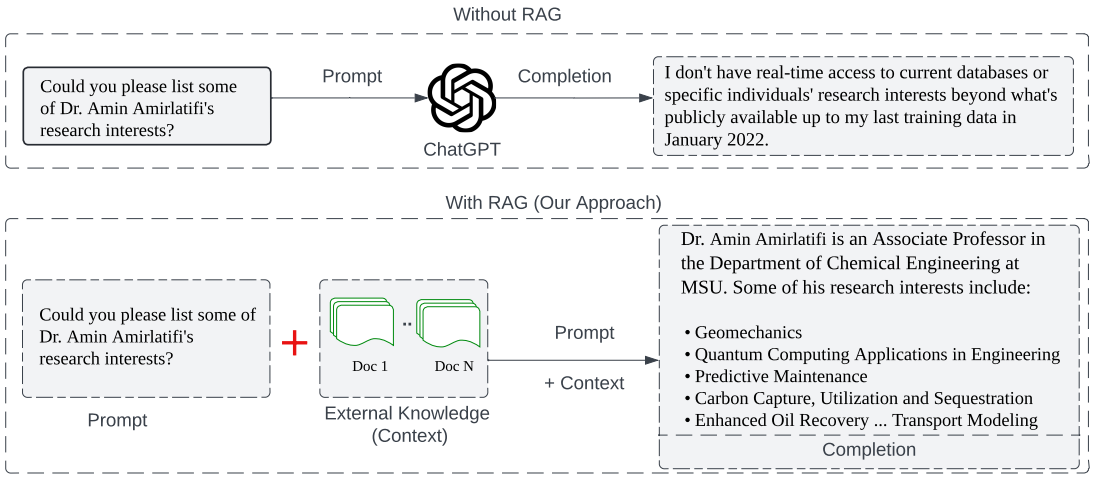


Fig. 1: Comparative example of completion (response generation) without using the RAG approach versus using the RAG approach for a given user prompt related to specific individual at Mississippi State University.

by Staffordshire University. Beacon offers personalized and responsive support, including information on timetables and answers to frequently asked questions.

Apart from the higher education sector, chatbots are being increasingly adopted across a diverse range of industries and contexts including healthcare [7], [8], cybersecurity [9], [10], retail [11], and hospitality [12] among others due to their ability to emulate human conversations, automate services, and reduce human workload. The meteoric rise in interest in using chatbots by industries at present is attributed to the overwhelming success of ChatGPT.[1]. In fact, the global chatbot market size was valued at 5.39 billion dollars in 2023 which is expected to reach 42.83 billion dollars by 2033, according to a market research report [13] published by Spherical Insights & Consulting.

In this paper, we introduce BARKPLUG V.2- the second iteration of a chatbot system built for Mississippi State University (MSU), with an architecture that can be applied to any university setting. This system acts as an interactive tool, capable of leveraging all university resources to provide more intelligent analyses of university related content. It helps users navigate university resources by directing them to relevant information. Compared to the other chatbots in the same area, BARKPLUG V.2 is more comprehensive, covering several aspects of university functions and services. The development of our chatbot utilizes Retrieval Augmented Generation (RAG) [14] techniques for response generation. RAG pipelines consist of two vital components: a retriever and a generator based on a Large Language Model (LLM). We opt for the RAG approach because pretrained LLMs, such as *gpt-3.5-turbo*, alone cannot adequately answer domain-specific questions or perform well on data outside their training dataset, often resulting in hallucinated outputs. Figure 1 provides a comparative overview of the response generation for a given user prompt without RAG and with RAG. As is evident from Figure 1, ChatGPT clearly fails to answer domain-specific questions, while BARKPLUG V.2, which uses the RAG approach, can accurately answer a user prompt. In our pipeline we utilize various campus resources such as information on *academic departments, financial aid, admission, scholarships, dining, housing, and health center* as a corpus of external data source for retrieval.

BARKPLUG V.2 project's key contributions include:

- Design and development of a comprehensive chatbot system proficient in responding to a wide spectrum of queries pertaining to the diverse array of campus resources available at MSU.
- Demonstrating the possibility of promptly providing personalized, real-time information, thereby augmenting user engagement through the continuous availability of the chatbot.
- Showcasing the application's effectiveness through rigorous evaluation, validating its performance and user satisfaction.

The rest of this article is divided into five connected sections. In Section II, we present the background and related work. Following that, in Section III, we explain the architecture and methodology. Section IV provides a detailed analysis of experimental results. Moving on to Section V, we provide implementation details and discuss the limitations and future works. Finally, we conclude our paper.

## II. BACKGROUND AND RELATED WORK

In this section, we briefly look into the pre-requisite background followed by exploring related research that focuses on development of chatbot applications in educational context.

### A. Large Language Models (LLMs)

Large Language Models (LLMs) like GPT-4, LLAMA3, and PaLM are at the forefront of computational linguistics, powered by Transformer-based architectures [15] with vast parameter spaces, often exceeding hundreds of billions. Transformers are neural network architectures that use self-attention mechanisms to efficiently process and generate sequences of data. LLMs models rely on the self-attention mechanism

[1]https://chatgpt.com/

within Transformers. LLMs excel in understanding and generating human language, reshaping the Natural Language Processing (NLP) landscape. They leverage various Transformer architectures and pre-training objectives, including decoder-only models (generates output based solely on a given context or prompt, without data transformation stage, e.g., GPT2, GPT3), encoder-only models (processes and transforms input data into a fixed representation, e.g., BERT, RoBERTa), and encoder-decoder architectures like BART.

These architectures efficiently process sequential data, capturing intricate dependencies within text while enabling effective parallelization. LLMs integrate prompting or in-context learning, enhancing text generation by incorporating contextual information. This capability facilitates coherent and contextually relevant responses, fostering interactive question-and-answer engagements [16].

### *B. Retrieval Augmented Generation (RAG)*

Pre-trained Large Language Models (LLMs) are proficient at acquiring extensive knowledge but lack memory expansion or revision capabilities, leading to errors like hallucinations. To address this, hybrid approaches like Retrieval Augmented Generation (RAG) have emerged [14], [17], [18].

RAG integrates input sequences with information retrieved from corpus of an external data source, enriching context for sequence generation. The retriever component selects the top $k$ text passages relevant to the input query, augmenting the model's understanding and enhancing output sequence generation. This process is governed by the equation: $p_n(z|x)$ where $p_n$ represents the retriever component with parameters (number of documents or passages a user wants to retrieve), selecting relevant passages $z$ from the knowledge database given input $x$.

### *C. Related Works*

Recent research on educational chatbots explores various areas such as application fields, objectives, learning experiences, design approaches, technology, evaluation methods, and challenges. Studies have shown that educational chatbots are used in health advocacy, language learning, and self-advocacy. They can be flow-based or powered by AI, facilitating answering Frequently Asked Questions (FAQs), performing quizzes, recommending activities, and informing users about various events [19] [20]. Chatbots have been found to improve students' learning experiences by motivating them, keeping them engaged, and providing immediate online assistance [21]. Additionally, chatbots make education more accessible and available [20]. Design aspects such as the role and appearance of chatbots are significant factors that affect their effectiveness as educational tools [22]. Chatbots are designed using various methods, including flow-based and AI-based approaches, and can incorporate speech recognition capabilities [23]. Technologies used to implement chatbots include Dialogflow [2] and ChatFuel[3] among others. These technologies impact chatbot performance and quality, necessitating careful selection during design and development [24]. Flow-based chatbots, such as those powered by Dialogflow, can provide structured interactions based on predetermined scripts, while AI-based chatbots leverage machine learning and NLP to offer more flexible and dynamic interactions.

In regards to assessment of the effectiveness of educational chatbots, evaluation methods such as surveys, experiments, and evaluation studies are used, measuring acceptance, motivation, and usability [25] [24] [26]. Surveys gather feedback from students and educators regarding their experiences with chatbots, while experiments may involve testing chatbots in controlled settings to measure their impact on learning outcomes. Evaluation studies provide deeper insights into how chatbots perform in various educational scenarios and how users perceive their usefulness. In terms of interaction styles, research examines whether chatbots are user-driven or chatbot-driven, depending on who controls the conversation [23] [19]. Chatbot-driven interactions often involve more automated and guided conversations, while user-driven interactions prioritize user input. Striking a balance between these approaches can result in more natural and effective communication. However, it's important to acknowledge that achieving this balance necessitates addressing substantive challenges to optimize the chatbot's applicability across diverse contexts, including the field of education.

Ethical considerations, such as compliance with educational norms and safeguarding user data, assume paramount importance [21], [27]. Leveraging novel methodologies in their development, we aim to navigate these issues more effectively. Moreover, we confront persistent programming complexities and the importance of sustaining chatbot utility amidst educational evolution [28], [29]. By harnessing advancements in technology, we endeavor to bolster our chatbots' resilience to these challenges. These collaborative endeavors offer a strategic direction, utilizing technological advancements to refine educational chatbots. Furthermore, the language model (conversational chatbot) contends with conceptual challenges essential for its operational efficacy, requiring careful research focus.

Insights from studies such as [30] reveal how language models such as BERT establish relationships between expressions and queries, shedding light on chatbot interaction styles and response quality. This study contributes to understanding how advanced language models can be integrated into chatbots for more nuanced and context-aware responses. [31] discusses the gap between chatbot responses and user intent, which can be more pronounced in complex university settings. Chatbots in academic environments often encounter questions that require a deep understanding of the subject matter and context. This necessitates the use of sophisticated models that can handle intricate queries and provide accurate and relevant responses. [30], [31] underscores the importance of understanding and controlling the context of language models, thereby guiding our efforts to integrate advanced language models into chatbots for more nuanced and context-aware responses. Their context-

[2]https://cloud.google.com/dialogflow

[3]https://chatfuel.com/

aware approach has been instrumental in shaping our chatbot's unique capabilities.

The integration of chatbots within university platforms and metaverses offers promising avenues for enhancing user experience and facilitating learning. For instance, [32] demonstrate how chatbots in metaverse-based university platforms offer instant, personalized support for tasks such as course navigation and answering FAQs, leveraging NLP and machine learning to streamline information dissemination and reduce administrative burdens. This kind of integration not only facilitates academic processes but also helps in addressing students' concerns promptly, ensuring smoother academic experiences. In specific university contexts, [33] develops a question-answering system for an Indonesian university admissions using Sequence-to-sequence learning. This system demonstrates how chatbots can be employed in specialized areas to address particular challenges, such as providing guidance during the admissions process. Similarly, [34] introduce a dynamic chatbot enhancing student interaction by covering admissions, academic assistance, and event information, prioritizing user feedback for accuracy, reliability, and safety. Frequent updates ensure that chatbots maintain relevance and continue to serve as effective tools for student support. Moreover, [35] presents TutorBot+, which employs LLMs like ChatGPT to offer feedback in programming courses. Their quasi-experimental research shows positive impacts on students' computational reasoning abilities, illustrating the potential of such interventions in education. TutorBot+ demonstrates the benefits of integrating advanced AI models to support students in understanding complex programming concepts, potentially transforming how computational subjects are taught.

## III. BARKPLUG V.2 ARCHITECTURE & METHODOLOGY

This section describes the architecture of BARKPLUG V.2 consisting two main phases: *context retrieval* and *completion* as shown in Fig. 2. The first phase retrieves documents relevant to the user prompt. The second phase utilizes these retrieved documents and user prompts to generate contextual responses referred to as completions. The subsequent subsections will provide a comprehensive breakdown of each phase, discussing their functionalities and methodologies.

### A. Context Retrieval

Retrieval in BARKPLUG V.2 involves obtaining pertinent information from an external data source to establish context for completions. This phase takes a prompt (query) as an input and produces some smaller, manageable sections or segments of text called chunks of documents that are relevant to the prompt. In our context, the external data is the university resources available through Mississippi State University's Website [4]. We curate data of 42 different department within the university using web crawlers. These include *academic departments, financial aid, admissions, housing, dinning services, library, health center* etc. Inclusion of these campus resources as external data source is to ensure BARKPLUG V.2 is comprehensive enough to answer diverse question. For example, a user might ask a question such as *"What are the funding opportunities available for graduate students in the CSE department?"*. Followed by the question *"Who do I contact if I have additional questions about majors or attending MSU?"* To answer the first question the system should have information about funding opportunities within CSE, whereas to answer the second question, information about *academic counselors* should be present in the external data source. Please refer to Section IV-A for detailed explanation on how we curate these data, prepare and process for this phase.

The first step in this phase is to transform the external data source. This step relies on two important components: an *embedding model* and a *vector database*. Embeddings refers to functions that map or transforms raw input data to low-dimensional vector representations while retaining important semantic information about the inputs [36]. On the other hand, the vector database is a type of database that stores data as high-dimensional vectors that are usually generated by applying embedding functions to the raw data [37], such as text in our case. It supports complex and unstructured data and allows fast and accurate similarity search and retrieval. BARKPLUG V.2 utilizes an embedding model to vectorize the external data sources, in particular, we leverage a *text-embedding-3-large* model managed through API calls. These vectors are subsequently stored in Chroma DB [38] an in-memory vector database.

For efficient context retrieval process we use vector store-backed retriever technique provided by LangChain [39]. It utilizes vector store to retrieve documents. In general the vector store retriever uses two types of search methods including Maximum Marginal Relevancy (MMR) and Similarity Search. In this work, we have leveraged *similarity score threshold retrieval (n)*, depicted in Fig.3 as our searching mechanism. This search strategy returns all possible results for a user prompt that meet a minimum similarity threshold. In our case, we have set this threshold to 0.2. The output of this phase is the relevant documents that serves as context for subsequent completions phase.

### B. Completion

The second phase is the *completion* which is also referred to as *response generation*. We utilize a gpt based LLM for completions, in particular, we leverage OpenAI's *gpt-3.5-turbo* as our base generator model. The input to the generator consists of retrieved document chunks and the user prompt. Then, the generator model, or LLM, uses this information as a guideline to produce acurate and relevant responses (completions). Response generation in BARKPLUG V.2 is managed through OpenAI's API calls.

An example of completion for the user prompts *"What are the funding opportunities available for graduate students in the CSE department?"* and *"Who do I contact if I have questions about attending MSU* can be seen in Fig. 4. In the first example, a user seeks information about funding

[4]https://www.msstate.edu/

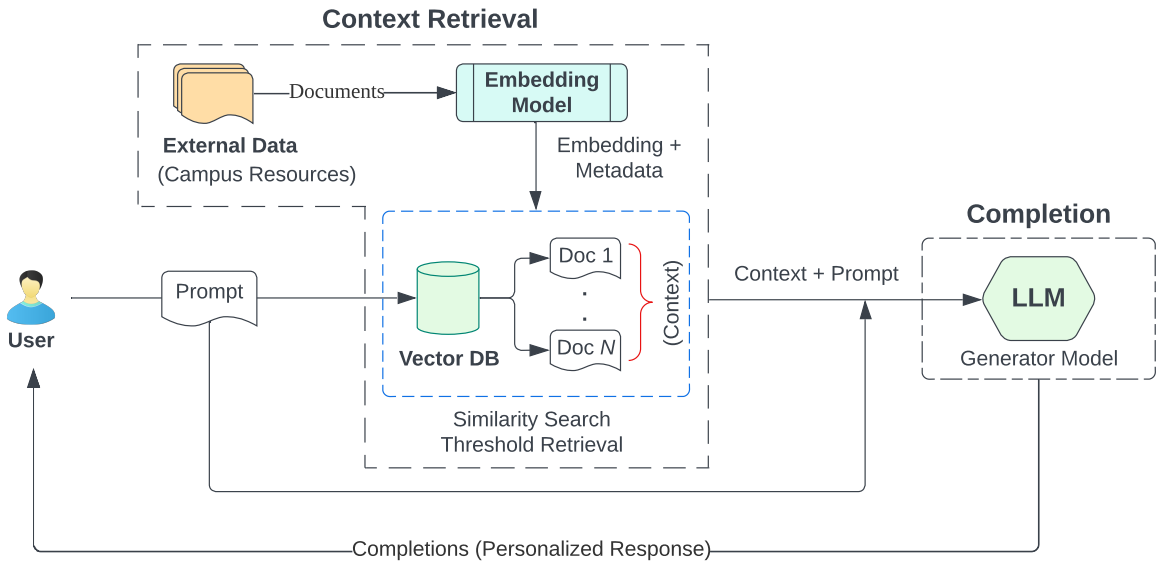


Fig. 2: Overview of BARKPLUG V.2's two phase architecture. The first phase *Context retrieval* is responsible to retrieve relevant documents based on the user prompt. The second phase, *Completion* responsible of generating personalized responses utlilizing retrieved documents as context along with user prompt.

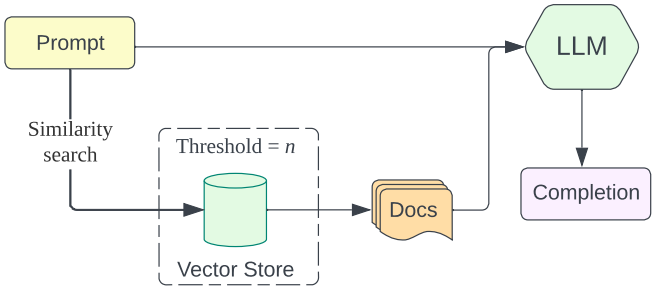


Fig. 3: Similarity score threshold retrieval.

opportunities within the CSE department for graduate studies. The completion, in this case, is a contextualized response generated by combining relevant information retrieved from campus resources, such as the academic department (CSE), with the user prompt. In the second example, the user is seeking information about admission to MSU. To answer this, the system must possess knowledge about campus resources, such as admissions and scholarships. As mentioned earlier, BARKPLUG V.2 not only has knowledge about academic departments but also other campus resources, including admissions and scholarships. By utilizing admissions and scholarships as external data sources, the system returns the relevant document reflecting information about admissions and scholarships, along with contact details, as a completion for the user prompt.

## IV. EXPERIMENTS AND RESULTS

To evaluate the performance of our system, we adopt a two-fold approach including both *quantitative* and *usability assessment* methods. For the quantitative evaluation (See Section IV-B), we utilize the RAGAS [40] framework, while the System Usability Scale (SUS) [41] is adopted for usability assessment (See Section IV-C). In the following subsection, we first discuss the dataset and steps we took to pre-process them and then provide a detailed explanation of our evaluation approaches.

**Prompt**: What are the funding opportunities available for graduate students in the CSE department?

**Completion**: Graduate students in the Computer Science and Engineering (CSE) department at [institution redacted] have access...funding opportunities...graduate assistantships, research grants, fellowships, scholarships, and industry partnerships...research grants cover project expenses.

**Prompt**: Who do I contact if I have questions about attending [institution redacted]?

**Completion**: For queries on admissions at [institution redacted], reach out to the Office of Admissions and Scholarships at phone number redacted or [email redacted].

Fig. 4: A graphical depiction of response generation. Red highlights indicate BARKPLUG V.2's ability to contextualize and generate factually grounded response.

### A. Dataset Description & Preparation

To ensure a comprehensive chatbot system capable of answering diverse questions—whether academic or non-academic—we initially developed a web scraper to gather information on various campus resources at Mississippi State University. This collection would then serve as an external data source in our pipeline. We scraped various campus resources including academic departments, financial aid, scholarships,

TABLE I: A subset of an external data source containing campus resources, including both academic and non-academic departments, indicating the total number of tokens associated with each.

| Campus Resources | **Departments** | **# of Tokens** |
|---|---|---|
| | Computer Science and Engineering | 200623 |
| | Chemical Engineering | 118271 |
| | Electrical and Computer Engineering | 328558 |
| | Industrial and Systems Engineering | 22390 |
| | Agricultural and Biological Engineering | 79978 |
| | Civil and Environmental Engineering | 61071 |
| | Aerospace Engineering | 37812 |
| | Biomedical Engineering | 256761 |
| | Housing | 132193 |
| | Admission | 276972 |
| | MSU Police | 16629 |

housing, dining, parking, and police. In total, we scraped 42 campus resources into a JSON file. Each JSON file includes the following information: the URL, title, and content of the scraped webpage, all wrapped into a JSON object. We consolidated the individual files into a master JSON file which serves as an external data source and is ingested into our RAG pipeline. A subset of the data utilized by BARKPLUG V.2 can be observed in Table I.

To enhance retrieval accuracy, we first preprocess the JSON file. This preprocessing step involves removing noise, such as undesirable Unicode characters, redundant, and unnecessary information. We then implement a recursive chunking strategy, with a chunk size of 8000 and an overlap of 1200 characters. This step is crucial for optimizing the performance of RAG chatbot systems with the objective of ensuring that our chatbot generates an accurate response that is contextually appropriate. Subsequently, we transformed the textual data into vectorized representations utilizing an *embedding model* (Refer to Section III-A to learn for more details on embedding models.).

### *B. Quantitative Evaluation*

To evaluate BARKPLUG V.2's ability to produce contextually appropriate responses, we utilize the RAGAS framework [40]. We choose this framework because it is specifically designed to assess RAG pipelines. Other popular evaluation metrics such as ROUGE [42] and BLEU [43] are not suitable in our context. This is because ROUGE is generally used to evaluate summarization tasks, while BLEU is designed to evaluate language translation tasks.

We evaluate both phase of BARKPLUG V.2 architecture (See section III) i.e. *context retrieval* and *completion*. To evaluate the retrieval, we employ two metrics such as *context precision and context recall*. The first metric represents the Signal-to-Noise Ratio (SNR) of retrieved context, while the second metric evaluates whether the retriever has the ability to retrieve all the relevant evidence to answer a question. Similarly, to evaluate *completion or generation* we employ *faithfullness* and *answer relevance* metrics. Faithfulness metric determine the extent to which the generated response relies solely on the provided context. The metric is between 0 and 1, lower the score less trustworthy the response. For example, if a user asks, "How can I contact Dr. Rahimi?" and the context states, "Dr. Shahram Rahimi is a professor at MSU...email address is rahimi@cse.msstate.edu," a faithful response would be, "You can contact Dr. Rahimi via his email rahimi@cse.msstate.edu.". Answer relevance, on the other hand, measures how pertinent the generated response is to the given question. When asked the question, "Could you please list some of Dr. Amin Amirilattafi's research interests?" the relevant context provided is a list of Dr. Amirilattafi's research areas: Geomechanics,..,Compressed Air Energy Storage, Big Data Analytics, Intelligent Fields,...and Transport Modeling. A relevant response to this question would be, "Dr. Amin Amirilattafi's research interests include Geomechanics, Quantum Computing Applications in Engineering, Predictive Maintenance, and Carbon Capture, Utilization, and Sequestration." The final RAGAS score, representing the harmonic mean of these four metrics, falls within a range of 0 to 1, with 1 denoting optimal generation. This score serves as a singular measure of a QA system's performance. Therefore, the RAGAS score is essential for assessing the overall performance and relevance of BARKPLUG V.2 in its targeted educational environments.

To conduct phase wise evaluation, we first crafted a set of questions and their ground truth pertaining to *engineering programs, general inquiries, research opportunities*, and *other university resources*. We report RAGAS score of 0.97, 0.96, 0.97 and 0.97 for these categories respectively in Table II. These score underlines both retrieval and completion component are efficient.

We also conduct end-to-end evaluation to measure overall performance of BARKPLUG V.2, as it directly affects the user experience. Metrics such as *answer similarity* and *answer correctness* are employed to assess the overall performance, ensuring a comprehensive evaluation. In particular, *answer similarity* (a metric that measures the semantic resemblance between the generated answer and the ground truth answer.) scores that reflect strong alignment with ideal responses are reported to be high in cases when questions about engineering programs and research opportunities are asked, with scores of 0.8434 and 0.8317 respectively. Moreover, *answer correctness* (a metric that determines how factually correct is the generated output.), which indicates high factual accuracy, is reported to be high when the system is asked questions about university resources and research opportunities, at 0.8923 and 0.8841 respectively. Overall, these metrics suggest that BARKPLUG V.2 effectively retrieves relevant and accurate answers.

### *C. Usability Assessment*

To further understand the user experience when using BARKPLUG V.2, we conducted a subjective satisfaction survey using the System Usability Scale (SUS) [41]. SUS is a widely reliable method that accesses systems usability through a set of 10-itemquestionnaire. Users rate their experience on a 5-point scale, and the resulting SUS score, calculated out of

TABLE II: Overview of results: Retrieval scores pertain to the *context retrieval* phase of the architecture, where *prec.* refers to context precision, and recall refers to context recall. Generation scores pertain to the *completion phase*, where *faith* stands for faithfulness and *rel.* for answer relevancy. The end-to-end evaluation showcases BARKPLUG V.2's efficiency in generating contextually relevant and accurate answers through metrics such as answer similarity and answer correctness.

| | Retrieval | | Generation | | RAGAS Score | End-to-End Evaluation | |
|---|---|---|---|---|---|---|---|
| Category | Prec. | Recall | Faith. | Rel. | Harmonic Mean | Answer Similarity | Answer Correctness |
| Engineering Programs | 0.98 | 0.96 | 0.99 | 0.97 | 0.97 | 0.8434 | 0.8620 |
| General Inquiry | 0.95 | 0.97 | 0.98 | 0.96 | 0.96 | 0.7764 | 0.8123 |
| Research Opportunities | 0.97 | 0.98 | 0.96 | 0.99 | 0.97 | 0.8245 | 0.8841 |
| University Resources | 0.96 | 0.99 | 0.97 | 0.98 | 0.97 | 0.8317 | 0.8923 |

100, reflects perceptions of ease of use, efficiency, and overall satisfaction. Given the expensive nature of this evaluation we engage a panel of 50 graduate and undergraduate students undertaking CSE8011 (Seminar course) at Mississippi State University. The participants were tasked to answer a set of 10 questions as depicted in Fig. 5, each offering five response options ranging from "strongly agree" to "strongly disagree". We then collected their feedback and calculated an average SUS score of 67.75. The feedback results indicated satisfactory usability with a room for improvement for future iterations of our system.

## V. DETAILED ANALYSIS AND INSIGHTS

In this section, we discuss implementation details, where we explain the technical process behind developing BARKPLUG V.2. Then, we discuss constraints and shortcomings encountered, and provide our plan for the future.

### A. Implementation Detail

For data curation we employed a multi-thread web crawler with the Scrapy Python library to collect data from over 42 campus resources (See Section IV-A for details). We carefully selected important HTML div tags that comprised of relevant information about a topic. This process was semi-automatic in nature because every HTML pages were differently formatted with different div ids. Manual div selection also allowed us to remove noise to some extent. The data was exported to JSON file format with url, topic and content. Individual JSON files for each of the campus resources was then consolidated into a master JSON file for comprehensive retrieval.

We predominantly use LangChain framework to develop BARKPLUG V.2. First, we preprocess master JSON into smaller chunks using Recursive Character Text Splitter splitting strategy. Given the nature of our data we opted for 8000 chunk size with 1200 overlap. We then apply an embedding function on these chunks utilizing OpenAI's *text-embedding-3-large* model and store the vectors in Chroma DB. This step allowed us to retrieve documents relevant to specific user prompts. In our case, we utilize *vectorstore* for *context retrieval* with a similarity search threshold as our search strategy (See Section III-A for more details). For completion or response generation we leverage OpenAI's *gpt-3.5-turbo* model. Both embedding and response generation is managed through API calls.

BARKPLUG V.2 is built with Django framework using python. For front-end we utilize HTML, CSS and Javascript. The current version of our system has not only question-answering functionality but also user sign up and log in feature. Once a user is registered they can ask queries, they can see previous conversations, delete conversations, and email conversations. Our application is deployed through Amazon Web Services (AWS), utilizing its scalable cloud infrastructure to ensure robust performance and accessibility. We use Docker for containerization, which allows us to package the application with all its dependencies, providing a consistent and reliable deployment environment across different systems.

### B. Limitations & Future Direction

Despite the achievements in developing our educational chatbot, several significant challenges currently limit its broader application.BARKPLUG V.2 does not currently have Automatic Speech Recognition (ASR) capability, which might hinder its use among visually impaired, disabled, or elderly users. Additionally, given that MSU hosts a number of international students annually from non-English speaking countries, it currently lacks multi-lingual support. In terms of technical limitations, our retrieval system sometimes fails to provide accurate or relevant results, occasionally producing incorrect information, a phenomenon known as 'hallucinations'. We are also limited by a maximum number of output tokens, which is 4096, and a context window of 16k. This sometimes hinders system's ability to capture the full length of the conversation in the memory buffer.

To address the limitations discussed above and enhance BARKPLUG V.2's functionality and usability, we are planning several key upgrades. These include adding support for multiple languages to cater to a diverse user base, integrating ASR and text conversion features to enable various interaction modes, and improving the retrieval algorithms to boost the accuracy and relevance of the information provided. Moreover, in response to the token limitations of the OpenAI API, we aim to apply the map-reduced document chain approach from LangChain. Through these improvements, we aim to transform BARKPLUG V.2 into a more reliable and accessible educational tool.

## VI. CONCLUSION

This study highlights the significant potential of AI-based chat systems in improving communication and access to in-

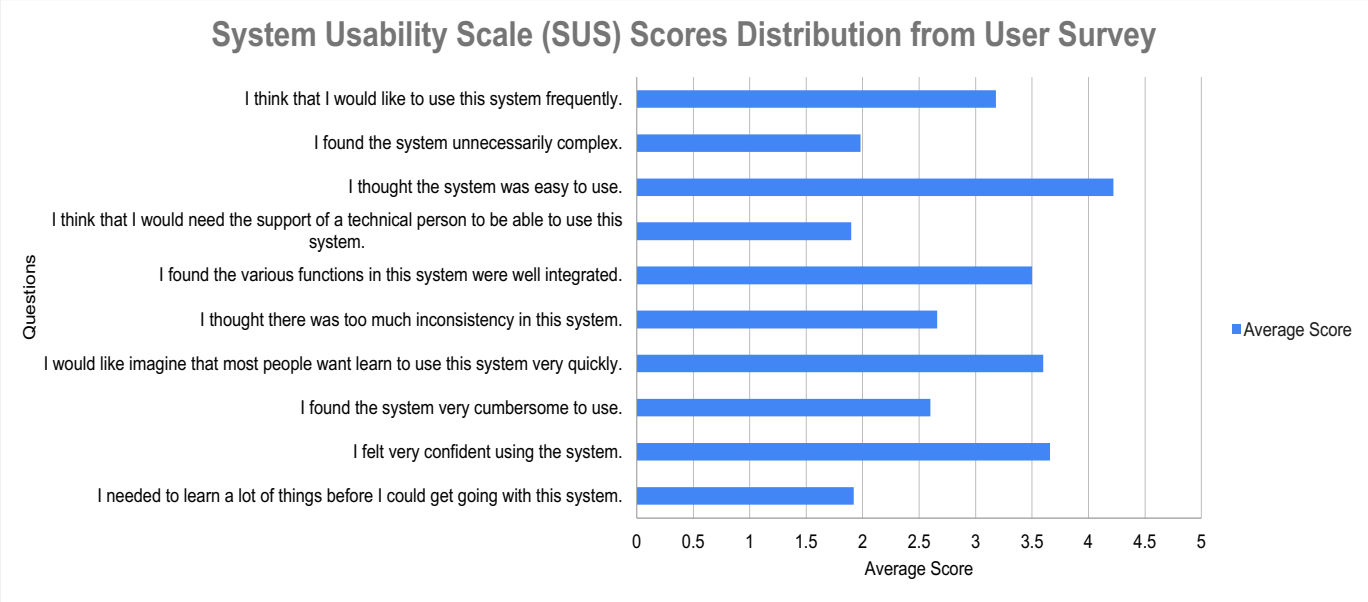


Fig. 5: Distribution of average System Usability Scale (SUS) scores.

formation regarding university resources. Our system, BARKPLUG V.2 integrates large amounts of university data, including academic programs, campus facilities, student service as external data corpus into its RAG pipelines for domain-specific question and answering tasks. By incorporating this external data corpus, our system ensures the delivery of precise and contextually relevant responses to both academic and non-academic user inquiries. The comprehensive end-to-end evaluation process demonstrated BARKPLUG V.2's efficiency in generating contextually relevant and accurate answers as measured by metrics such as answer similarity and correctness. Furthermore, system usability experiments employing the SUS indicated that BARKPLUG V.2 is practical and effective for real-world usage, affirming its reliability and the positive user experience it offers. The positive outcomes of using BARKPLUG V.2 at MSU suggest promising opportunities for broader implementation. This system could be adapted for use in other universities or different sectors and can be viewed as enterprise document retrieval systems that enhance user engagement and information access.

## DECLARATION OF COMPETING INTEREST

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

## ACKNOWLEDGEMENT

This work was supported by the PATENT Lab (Predictive Analytics and Technology Integration Laboratory) at the Department of Computer Science and Engineering, Mississippi State University.

## REFERENCES

[1] K. A. Meyer and S. Jones, "Information found and not found: what university websites tell students," *Online journal of distance learning administration*, vol. 14, no. 3, pp. 1–10, 2011.

[2] C. Khatri, A. Venkatesh, B. Hedayatnia, R. Gabriel, A. Ram, and R. Prasad, "Alexa prize—state of the art in conversational ai," *AI Magazine*, vol. 39, no. 3, pp. 40–55, 2018.

[3] L. Bradeško and D. Mladenić, "A survey of chatbot systems through a loebner prize competition," in *Proceedings of Slovenian language technologies society eighth conference of language technologies*, vol. 2. sn, 2012, pp. 34–37.

[4] ASU. Introducing sunny. [Online]. Available: https://heysunny.asu.edu/about

[5] GSU. Reduction of summer melt. [Online]. Available: https://success.gsu.edu/initiatives/reduction-of-summer-melt/

[6] S. University. Beacon - your digital guide. [Online]. Available: https://www.staffs.ac.uk/students/digital-services/beacon

[7] L. Laranjo, A. G. Dunn, H. L. Tong, A. B. Kocaballi, J. Chen, R. Bashir, D. Surian, B. Gallego, F. Magrabi, A. Y. Lau *et al.*, "Conversational agents in healthcare: a systematic review," *Journal of the American Medical Informatics Association*, vol. 25, no. 9, pp. 1248–1258, 2018.

[8] S. Neupane, S. Mitra, S. Mittal, N. A. Golilarz, S. Rahimi, and A. Amirlatifi, "Medinsight: A multi-source context augmentation framework for generating patient-centric medical responses using large language models," *arXiv preprint arXiv:2403.08607*, 2024.

[9] M. F. Franco, B. Rodrigues, E. J. Scheid, A. Jacobs, C. Killer, L. Z. Granville, and B. Stiller, "Secbot: a business-driven conversational agent for cybersecurity planning and management," in *2020 16th international conference on network and service management (CNSM)*. IEEE, 2020, pp. 1–7.

[10] S. Mitra, S. Neupane, T. Chakraborty, S. Mittal, A. Piplai, M. Gaur, and S. Rahimi, "Localintel: Generating organizational threat intelligence from global and local cyber knowledge," *arXiv preprint arXiv:2401.10036*, 2024.

[11] M. Chung, E. Ko, H. Joung, and S. J. Kim, "Chatbot e-service and customer satisfaction regarding luxury brands," *Journal of Business Research*, vol. 117, pp. 587–595, 2020.

[12] X. Y. Leung and H. Wen, "Chatbot usage in restaurant takeout orders: A comparison study of three ordering methods," *Journal of Hospitality and Tourism Management*, vol. 45, pp. 377–386, 2020.

[13] S. I. LLP. Global chatbot market size. [Online]. Available: https://finance.yahoo.com/news/global-chatbot-market-size-exceed-080000758.html

[14] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[16] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang *et al.*, "A survey on evaluation of large language models," *ACM Transactions on Intelligent Systems and Technology*, vol. 15, no. 3, pp. 1–45, 2024.

[17] F. Petroni, T. Rocktäschel, P. Lewis, A. Bakhtin, Y. Wu, A. H. Miller, and S. Riedel, "Language models as knowledge bases?" *arXiv preprint arXiv:1909.01066*, 2019.

[18] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung, "Survey of hallucination in natural language generation," *ACM Computing Surveys*, vol. 55, no. 12, pp. 1–38, 2023.

[19] S. Cunningham-Nelson, W. Boles, L. Trouton, and E. Margerison, "A review of chatbots in education: practical steps forward," in *30th annual conference for the australasian association for engineering education (AAEE 2019): educators becoming agents of change: innovate, integrate, motivate*. Engineers Australia, 2019, pp. 299–306.

[20] S. Wollny, J. Schneider, D. Di Mitri, J. Weidlich, M. Rittberger, and H. Drachsler, "Are we there yet?-a systematic literature review on chatbots in education," *Frontiers in artificial intelligence*, vol. 4, p. 654924, 2021.

[21] C. W. Okonkwo and A. Ade-Ibijola, "Chatbots applications in education: A systematic review," *Computers and Education: Artificial Intelligence*, vol. 2, p. 100033, 2021.

[22] A. S. D. Martha and H. B. Santoso, "The design and impact of the pedagogical agent: A systematic literature review." *Journal of educators Online*, vol. 16, no. 1, p. n1, 2019.

[23] R. Winkler, S. Hobert, A. Salovaara, M. Söllner, and J. M. Leimeister, "Sara, the lecturer: Improving learning in online education with a scaffolding-based conversational agent," in *Proceedings of the 2020 CHI conference on human factors in computing systems*, 2020, pp. 1–14.

[24] J. Q. Pérez, T. Daradoumis, and J. M. M. Puig, "Rediscovering the use of chatbots in education: A systematic literature review," *Computer Applications in Engineering Education*, vol. 28, no. 6, pp. 1549–1565, 2020.

[25] S. Hobert and R. Meyer von Wolff, "Say hello to your new automated tutor–a structured literature review on pedagogical conversational agents," 2019.

[26] G.-J. Hwang and C.-Y. Chang, "A review of opportunities and challenges of chatbots in education," *Interactive Learning Environments*, vol. 31, no. 7, pp. 4099–4112, 2023.

[27] K.-J. Tokayev, "Ethical implications of large language models a multidimensional exploration of societal, economic, and technical concerns," *International Journal of Social Analytics*, vol. 8, no. 9, pp. 17–33, 2023.

[28] E. Adamopoulou and L. Moussiades, "An overview of chatbot technology," in *IFIP international conference on artificial intelligence applications and innovations*. Springer, 2020, pp. 373–383.

[29] ——, "Chatbots: History, technology, and applications," *Machine Learning with applications*, vol. 2, p. 100006, 2020.

[30] H. Tripathi, "Experimental approach toward training and analysing siamese deep neural network for sentence with no repeated expressions," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. IEEE, 2023, pp. 1–5.

[31] M. A. Kuhail, N. Alturki, S. Alramlawi, and K. Alhejori, "Interacting with educational chatbots: A systematic review," *Education and Information Technologies*, vol. 28, no. 1, pp. 973–1018, 2023.

[32] Q. Xie, W. Lu, Q. Zhang, L. Zhang, T. Zhu, and J. Wang, "Chatbot integration for metaverse-a university platform prototype," in *2023 IEEE International Conference on Omni-layer Intelligent Systems (COINS)*. IEEE, 2023, pp. 1–6.

[33] Y. W. Chandra and S. Suyanto, "Indonesian chatbot of university admission using a question answering system based on sequence-to-sequence model," *Procedia Computer Science*, vol. 157, pp. 367–374, 2019.

[34] P. F. Oliveira and P. Matos, "Introducing a chatbot to the web portal of a higher education institution to enhance student interaction," *Engineering Proceedings*, vol. 56, no. 1, p. 128, 2023.

[35] C. Martinez-Araneda, M. Gutiérrez, D. Maldonado, P. Gómez, A. Segura, and C. Vidal-Castro, "Designing a chatbot to support problem-solving in a programming course," in *INTED2024 Proceedings*. IATED, 2024, pp. 966–975.

[36] C. Song and A. Raghunathan, "Information leakage in embedding models," in *Proceedings of the 2020 ACM SIGSAC conference on computer and communications security*, 2020, pp. 377–390.

[37] Y. Han, C. Liu, and P. Wang, "A comprehensive survey on vector database: Storage and retrieval technique, challenge," *arXiv preprint arXiv:2310.11703*, 2023.

[38] Chroma. Chroma: The ai-native open-source embedding database. [Online]. Available: https://www.trychroma.com/

[39] LangChain. Applications that can reason. powered by langchain. [Online]. Available: https://www.langchain.com/

[40] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "Ragas: Automated evaluation of retrieval augmented generation," *arXiv preprint arXiv:2309.15217*, 2023.

[41] P. Vlachogianni and N. Tselios, "Perceived usability evaluation of educational technology using the system usability scale (sus): A systematic review," *Journal of Research on Technology in Education*, vol. 54, no. 3, pp. 392–409, 2022.

[42] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.

[43] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.