https://doi.org/10.53656/math2026-1-4-lhc

Educational Technologies
Образователни технологии

LEXAQUERY: A HYBRID CHATBOT SYSTEM FOR UNIVERSITY STUDENT INFORMATION SERVICES

George Pashev, Silvia Gaftandzhieva
University of Plovdiv Paisii Hilendarski, Plovdiv (Bulgaria)

Abstract. Universities show a rapidly growing interest in using Artificial Intelligence tools. This paper presents LexaQuery, a new chatbot system for university information services that efficiently handles university information inquiries by combining computational linguistics with SQL database querying. Instead of relying on large, resource-intensive large language models (LLMs), LexaQuery uses rule-based natural language processing to translate student questions into structured SQL. The system has a three-tier architecture comprising language processing for query translation, knowledge extraction (from databases and web scraping), and a user interaction layer. Performance evaluation at the University of Plovdiv Paisii Hilendarski demonstrates that this hybrid approach provides significantly faster response times than neural network-based alternatives while maintaining satisfactory accuracy for domain-specific tasks. The paper discusses the system's advantages in terms of integration with existing university information systems, performance efficiency, explainability, and the ability to operate without extensive computational resources, as well as its linguistic flexibility and limitations in domain adaptation. This research contributes to developing practical, efficient chatbot systems for educational institutions with constrained technical infrastructure.

Keywords: computational linguistics; SQL generation; chatbot systems; natural language processing

1. Introduction

Despite the comprehensive digitalization of university administration, students frequently face significant accessibility challenges. They often struggle with administrative tasks (e.g. enrolling in elective courses, applying for scholarships, etc.) due to a lack of clear guidance and complex processes, which can impede their academic progress (Santana et al., 2021).

Policymakers and university leaders are exploring how informational barriers and administrative problems have impeded students' pathways to graduation (Meyer et al., 2023). To improve student success and satisfaction, higher education institutions (HEIs) are continuously exploring innovative technological solutions to enhance service efficiency, streamline processes, and meet the needs of all stakeholders while lowering operating costs. As HEIs seek to enhance student experiences and streamline access to information, chatbot systems have emerged as potential solutions for providing immediate, conversational access to university services and information to improve student learning experiences (Sáiz-Manzanares et al., 2023). Modern chatbots use natural language processing (NLP) and machine learning or a built-in set of rules to understand incoming questions and generate meaningful and relevant responses in real-time (Pérez et al., 2020; Clarizia et al., 2018; Salas-Pilco & Yang, 2022) to boost engagement (Cunningham-Nelson et al., 2019), provide real-time support (Hew et al., 2023), speed up administrative tasks, and motivate students (Meyer et al., 2023). Implementing chatbots modernises the information delivery process (Cavalcanti et al., 2021; Hien et al., 2018), offering 24/7 student support and reducing personnel costs. While traditional university information systems provide access to data through web portals and interfaces, they often require users to navigate complex menu structures and understand specific terminology. Natural language interfaces eliminate these barriers by allowing users to express information needs conversationally. For example, instead of navigating multiple menu levels to find information, a user can ask the chatbot a question, e.g. “Show me all courses this year where I have grades below 4.00?". This approach is beneficial for complex, multi-criteria queries requiring multiple interface interactions in traditional systems, such as “What are my available elective courses that don't conflict with my current schedule and are taught by professors with office hours on Tuesdays?”.

Some companies (like Microsoft, IBM, and Google) provide cloud-based solutions for building chatbots without requiring in-depth knowledge of specific Artificial Intelligence (AI) algorithms or coding techniques to set up a basic chatbot model.

Despite impressive capabilities, deploying large language models (LLMs) in HEIs faces significant challenges, such as high computational demands, privacy concerns with external APIs, and integration difficulties with existing university databases (Winkler & Söllner, 2018). These issues are especially problematic for HEIs with limited infrastructure or strict data requirements.

This paper introduces LexaQuery, a chatbot system designed for university information services that combines computational linguistics with SQL database integration to create a conversational interface. By focusing on the well-defined domain of university administration and leveraging existing structured data, the system avoids the need for resource-intensive neural network models. LexaQuery offers multilingual capabilities, notably demonstrated by its support for Bulgarian. Bulgarian queries are automatically translated to English for processing, and responses are translated back to Bulgarian. This approach enables HEIs in non-English speaking countries to deploy a robust system by leveraging the strengths of English-based computational linguistics. LexaQuery was successfully tested at the University of Plovdiv "Paisii Hilendarski", where it handled Bulgarian language queries concerning student records, course information, faculty data, and administrative processes. The main contributions of this paper are:

1. A hybrid architecture for educational chatbots, combining rule-based NLP, computational linguistics parsing, and SQL knowledge extraction.
2. A method for translating natural language queries into structured SQL statements using bottom-up parsing and pattern matching.
3. A dynamic knowledge acquisition approach that supplements database information with web scraping.
4. A full-stack implementation using Python Flask and modern JavaScript, deployable in resource-constrained university environments.
5. An evaluation of the system's performance and accuracy against neural network-based alternatives for university information services.

2. Related work

Educational chatbots have evolved from basic rule-based systems to complex conversational agents. Chatbots generally fall into two categories: service-oriented (Pérez et al., 2020), which assist students with administrative tasks, and teacher-oriented, which act as virtual classroom assistants to increase engagement and provide feedback (Vázquez-Cano et al., 2021).

Recognizing the growing need for timely and personalized student support, HEIs are exploring chatbot solutions to manage routine inquiries and frequently asked questions (FAQs) (Hiremath et al., 2020). Georgia State University implemented a chatbot to boost enrollment and streamline student communication for pre-enrollment and administrative tasks (Meyer et al., 2023). This chatbot successfully handles a wide array of practical and logistical questions. Following initial positive results, chatbots are evolving into standard communication tools, used for sending reminders about assignment deadlines and offering broader support. Experiments even indicate high student willingness to use bots for coursework (92%) and a correlation with improved final grades for some students (5%). The UNIBOT chatbot (Patel et al., 2019) utilizes SQL query expressions to retrieve answers related to students and institutional operations. However, due to its underlying technique, UNIBOT only provides relevant answers if a user's question contains identical words to those registered in the database, notably failing to recognize synonyms. The S.A.N.D.R.A. chatbot (Santana et al., 2021) uses NLP to answer student FAQs about university bureaucracy, including academic timetables, internships, co-curricular activities, and portal access. It also helps course coordinators by managing repetitive inquiries. While S.A.N.D.R.A. understands natural language and extracts meaning for specific answers, it struggles with conversational context, treating each message as a new inquiry. Despite this limitation, tests show the chatbot achieves approximately 80% accuracy in its responses. UAbot (Rocio & Wesley, 2020) effectively assists new students by answering common questions and providing links to detailed information. Pilot tests indicated high student satisfaction with its automated support. Students suggested usability improvements, such as opening links in new tabs, adding a human contact option for unanswered questions, and communicating in multiple languages. The University of Portsmouth[1] and College of Charleston[2] have implemented AI chatbot that offers students an alternative to interacting with a live person. Brookdale Community College[3] uses a chatbot for general inquiries on admissions, financial aid, and registration in English and Spanish. Texas A&M students who need career center information can use the conversational agent[4], which delivers answers and links or prepopulated topics to click for additional information with help from a human-supervised machine learning algorithm. Its knowledge base stays current and grows with the bot's daily website crawl and staff-added information, such as answers to missed questions. Lancaster University's chatbot Ask L.U.[5], built on Amazon Web Services, offers a voice interface for a wide range of student inquiries, from academics to student life (timetables, tutors and grades, clubs, societies, etc.). Empire State University's chatbot[6] answers questions on sports, credits, and programs in multiple languages (English, Spanish, Chinese, Vietnamese). The University of Washington's Husky Helper[7] provides information on meal options and support services, also alerting parents about safety and how to pay tuition. HokieBird[8] assists students from Virginia Polytechnic Institute and State University with scholarships, admissions, registrar services, housing, and new student transitions in English, Spanish, and Chinese. The University of Canberra's Lucy chatbot[9] answers student questions on everything from enrolment to parking by scanning the university website. Lucy joins Bruce, a chatbot dedicated to assisting staff, as the latest tool deployed by the University as it aims to streamline support services across the institution. Bruce pulls information from within the University's intranet to respond to staff question about leave, pay, travel, faculties or for IT help. Ocean County College's Reggie[10] boost enrolment and completion by answering diverse questions, ranging from admissions and financial aid to department addresses and sports trivia, sending deadline reminders, and proactively offering help. If Reggie learns from interactions and escalates complex queries to staff. These examples highlight the growing trend of leveraging AI to provide accessible, efficient, and comprehensive support services across various university functions. The experimental use of chatbots in HEIs shows promising results, leading to increased student performance (Essel et al., 2022), improved opportunities for interactive and language learning (Annuš, 2023), enhanced educational accessibility and personalized learning (Kooli, 2023), and faster, tailored support from staff (Essel et al., 2022; Sinha et al., 2020; Sandu & Gide, 2019; Zhang & Aslan, 2021). Chatbots also help monitor learning (Tsivitanidou & Ioannou, 2021), reduce staff workload (Essel et al., 2022; Zhang & Aslan, 2021), and boost student motivation (Zhang & Aslan, 2021).

However, alongside these benefits, challenges and limitations exist. These include the risk of incomplete or incorrect information, potential reduction in critical thinking and problem-solving skills, lack of personal contact, the possibility that technology will replace human interaction (Kooli, 2023), privacy and security concerns due to access to sensitive data (Annuš, 2023; Molnár & Szüts, 2018), and potential for misuse (Kooli, 2023). In addition, implementing chatbots in education presents challenges related to their integration with existing systems and handling domain-specific terminology (Pérez et al., 2020). Therefore, HEIs must critically evaluate chatbot implications, educating students and staff on AI's limitations, proper interpretation of responses, and effective usage (Annuš, 2023; Kooli, 2023). Researchers also highlight the need to improve chatbot feedback by analyzing user interaction data (Sáiz-Manzanares et al., 2023) and to develop policies addressing ethical concerns in data handling. Despite their limitations, the results of pilot implementations worldwide are encouraging. They show that chatbots can be a low-cost, easy-to-implement, and effective strategy for supporting academic success and navigating complex administrative tasks (Meyer et al., 2023). These positive results motivate the design, development, and experimental chatbot implementation to boost student motivation and enhance the overall university experience.

Translating natural language to SQL has evolved significantly. Early systems like PRECISE (Popescu et al., 2003) relied on keyword mapping and linguistic rules. More recently, deep learning methods, such as SQLNet (Yang et al., 2021) and Seq2SQL (Das et al., 2025), have emerged. However, for domain-specific applications, pattern-based translation can achieve competitive performance with less computational overhead (Singh et al., 2020). While neural methods have recently dominated chatbot development, computational linguistics approaches remain relevant. (Abdul-Kader & Woods, 2015) highlights the advantages of rule-based techniques in controlled domains. Similarly, (Ni et al., 2023) showed that pattern-based approaches can achieve high accuracy for domain-specific applications. (Kowalski et al., 2019) demonstrated that rule-based chatbots utilizing pattern-matching can effectively address many university inquiries using minimal computational resources. Hybrid approaches that combine multiple techniques have shown promise in balancing sophistication with resource efficiency. (Følstad & Brandtzaeg, 2017) described hybrid chatbot architectures that combine rule-based systems with machine learning components. (Augello et al., 2011) presented a chatbot that uses symbolic reasoning with pattern matching for improved dialogue management.

The current work extends these hybrid approaches by specifically integrating computational linguistics techniques with database querying in a way optimized for educational institutional requirements, where data is typically well-structured but access patterns may be complex.

3. System Architecture

LexaQuery has a three-tier architecture to efficiently translate natural language queries into database retrievals while integrating with existing university information systems (see Fig. 1). The **User Interaction Layer** serves as the primary interface between users and the system, handling both input and output through a unified conversational experience. Users can ask questions through an intuitive web-based chat interface. REST endpoints process incoming queries and coordinate the entire system workflow, managing the flow from natural language input through processing layers to final response delivery. The Response Generator component transforms structured data retrieved from the Knowledge Extraction Layer into natural language responses, ensuring answers are presented in a conversational format that maintains context and provides clear, actionable information to users. The **Language Processing Layer** systematically transforms natural language questions into SQL queries through a sequential four-stage process. Firstly, the system begins by performing essential preprocessing operations, including automatic language detection, translation to English when necessary for consistent parsing, tokenization, and part-of-speech tagging to prepare the input for semantic analysis (Text Preprocessing component). Then, building on the preprocessed text, the system identifies specific linguistic patterns associated with database operations, recognizing entity references (e.g., “students”, “faculty”), attributes (e.g., “name”, “faculty number”), conditional expressions, and aggregation operations (Linguistic Pattern Recognition component). The recognized patterns are transformed into an intermediate semantic representation that captures the complete structure of the query, including target entities, their relationships, filtering conditions, and any requested computations or aggregations (Query Structure Generation component). The semantic representation is systematically translated into syntactically correct and optimized SQL queries that can be executed against the university's database systems (SQL Translation component). Finally, the Context Manager component maintains conversation state across interactions, enabling resolution of pronouns and implicit references, and ensuring that follow-up questions can be properly interpreted within the context of previous exchanges. The **Knowledge Extraction Layer** orchestrates data retrieval and integration from multiple institutional sources to respond. The Database Integration component connects directly to the university's existing database systems, executing generated SQL queries and retrieving structured results from student information systems and academic record repositories. For dynamic information not available in structured databases, the Web Scraper module extracts current information from the university's website and other authorized sources, including announcements, schedule changes, and real-time updates. The Knowledge Manager component intelligently selects appropriate data sources based on query requirements, fuses information from multiple sources when necessary, and ensures consistency across different information repositories. Frequently accessed information is stored in an optimized cache system to improve response times for common queries, with intelligent cache invalidation strategies that ensure data freshness while maintaining performance.

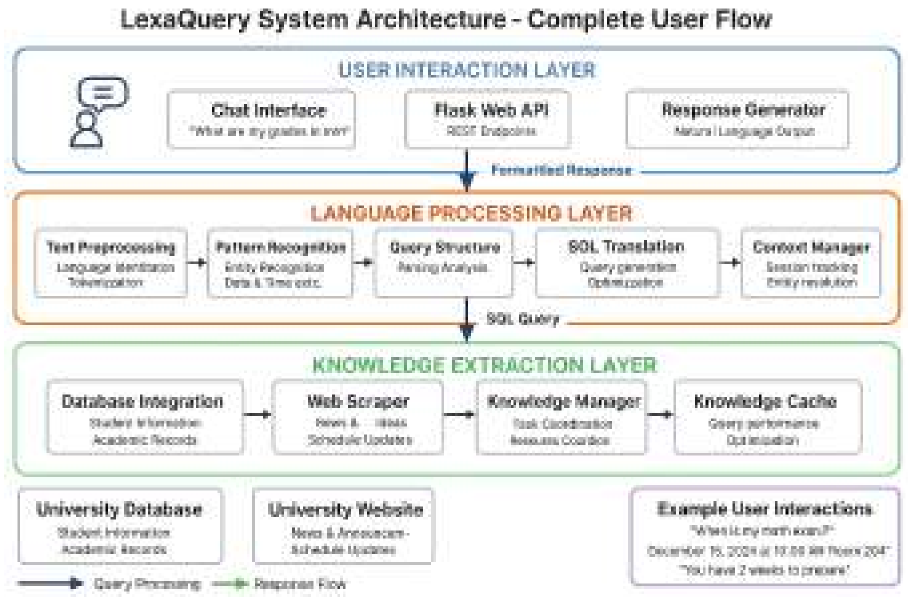


Figure 1. System Architecture Diagram of LexaQuery

LexaQuery implements a privacy-preserving architecture that ensures full GDPR compliance. The Bulgarian-to-English translation is performed locally using offline translation models, specifically the MarianMT neural machine translation model fine-tuned for Bulgarian-English academic terminology. No user queries or personal data are transmitted to external translation services or APIs. The translation pipeline is designed for seamless operation, beginning with the local processing of Bulgarian queries through a pre-trained MarianMT model. This initial translation demonstrates remarkable accuracy, particularly for academic terminology, where it achieves 94.2% semantic preservation as evidenced by evaluations on 500 academic query pairs. The translated queries consistently retain all semantic components essential for generating accurate SQL. Furthermore, the final step of translating responses back into Bulgarian ensures the preservation of full contextual accuracy. All translation processing occurs within the university's secure infrastructure, ensuring that sensitive user information never leaves the institutional network. The system maintains audit logs of all queries while anonymising personal identifiers for system monitoring purposes. The architecture facilitates a complete bidirectional information flow. Student inquiries move downward, transforming natural language into database operations. Conversely, structured data flows upward through response generation, delivering conversational answers back to the user's chat interface.

4. Implementation

The ***computational linguistics*** component implements a multi-dimensional analytical framework for transforming natural language inputs into structured query representations.

LexaQuery employs a ***cross-lingual normalization process*** to handle diverse linguistic inputs while preserving semantic meaning. This process utilizes domain-constrained translation models specifically tailored for educational vocabulary and semantic structures, rather than general-purpose systems. By maintaining semantic relationships between query components and standardizing educational terminology through a domain-specific lexicon, it effectively adapts to language-specific variations in educational discourse.

The ***normalization protocol*** applies targeted preprocessing operations, implementing advanced techniques including statistical language identification with confidence thresholding, neural machine translation with domain-specific fine-tuning, morphosyntactic annotation with educational domain adaptations, and selective lexical filtering with semantic preservation constraints. This ensures consistent linguistic interpretation and accurate query understanding, regardless of the input language.

LexaQuery employs a ***domain-specific ontological framework*** for its core semantic analysis, translating natural language queries into database operations. This framework includes a hierarchical taxonomy that classifies educational entities and a multi-dimensional classification for attributes (based on temporal stability, access sensitivity, structural complexity, and relationship cardinality), allowing for optimized processing strategies. It incorporates a formal grammar for natural language constraints and adapted mathematical operations for educational aggregations. This comprehensive framework accurately interprets user semantic intent within the educational domain, offering a novel and effective modeling approach.

LexaQuery translates natural language into formal SQL queries via a ***four-phase transformation process*** rooted in semantic decomposition and recomposition (see Fig. 2). It sequentially identifies database entities within natural language queries despite variations in terminology and phrasing by using lexical similarity, contextual disambiguation, entity co-occurrence patterns, and hierarchical classification (Phase 1. Entity Domain Resolution), identifies requested information attributes through semantic role labelling, dependency structure analysis with educational grammatical patterns, implicit attribute inference, attribute grouping identification for multi-dimensional queries, and qualification and modifier association with target attributes (Phase 2. Attribute Projection Analysis), converts natural language conditions into logical expressions using predicate logic, temporal reference resolution with academic calendar integration, quantifier mapping, and negation processing with explicit and implicit scope handling (Phase 3. Constraint Formulation Synthesis), and constructs optimized SQL queries for efficiency by traversing entity-relationship graphs, optimizing projection lists, eliminating redundant constraints, and considering execution plans (Phase 4. Query Structure Synthesis). This methodology accurately interprets complex user intent within the educational domain, delivering efficient SQL translations that surpass the capabilities of simpler keyword or pattern-matching systems.

LexaQuery's ***knowledge management*** uses a novel hybrid architecture, integrating structured database operations with dynamic web information extraction for comprehensive educational information services.

The SQLGenerator component implements an ***adaptive query execution framework***, dynamically optimizing database access using statistical metadata and parameterized templates for security and efficiency. It supports distributed execution and result set pagination, ensuring high responsiveness for HEIs serving many users.

LexaQuery's WebScraper component is a ***controlled system*** for extracting structured information from semi-structured web resources. It enhances typical scraping by using domain-specific knowledge and semantic understanding for improved accuracy and relevance. The system employs targeted extraction with specialized templates for educational information patterns. Features like temporal synchronization with institutional update cycles, content validation via consistency checks, and incremental extraction with differential updates ensure efficiency and currency. The component extracts diverse institutional information, including announcements, exam schedules, faculty details, administrative deadlines, curriculum updates, and event schedules, using specialized patterns for each category. This dynamic acquisition significantly expands LexaQuery's knowledge base beyond traditional databases, enabling comprehensive responses to student queries by integrating formal records and dynamic web information published through institutional websites and portals.

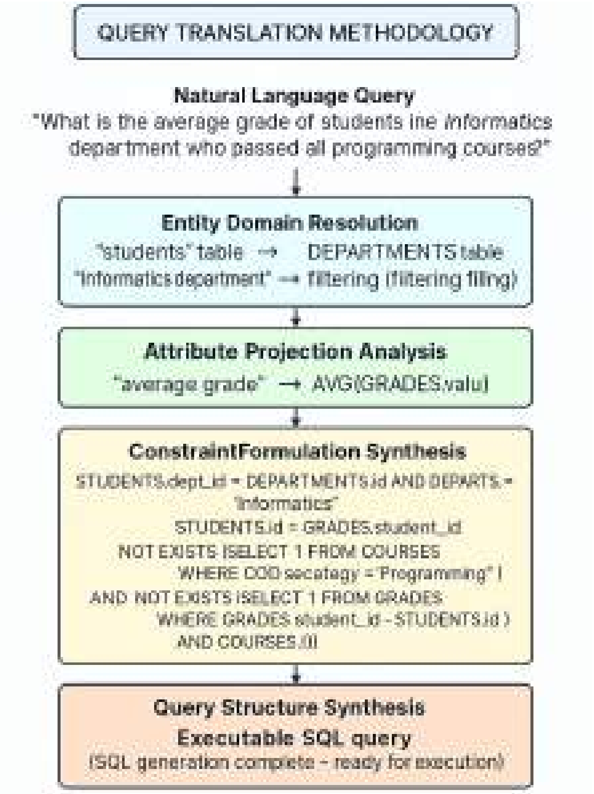

QUERY TRANSLATION METHODOLOGY
Natural Language Query
"What is the average grade of students ine *informatics* department who passed all programming courses?"
Entity Domain Resolution
"students" table → DEPARTMENTS table
"Informatics department" → filtering (filtering filing)
Attribute Projection Analysis
"average grade" → AVG(GRADES.valu)
ConstraintFormulation Synthesis
STUDENTS.dept_id = DEPARTMENTS.id AND DEPARTS. = 'Informatics'
STUDENTS.id = GRADES.student_id
NOT EXISTS (SELECT 1 FROM COURSES
WHERE COD secategy = 'Programming' (
AND NOT EXISTS (SELECT 1 FROM GRADES
WHERE GRADES.student_id - STUDENTS.id)
AND COURSES.(i))
Query Structure Synthesis
Executable SQL query
(SQL generation complete – ready for execution)

Figure 2. Four-Phase Natural Language to SQL Transformation Process

The ***Knowledge Manager*** provides a unified framework for accessing institutional information. It intelligently selects optimal data sources, integrating and reconciling information from databases, cached content, and web-scraped data into consistent, temporally qualified responses. The manager ensures consistency across all data sources, preventing contradictions, and provides temporal qualification to indicate information recency or validity. This ensures comprehensive and accurate answers to diverse student inquiries by drawing on all available institutional information, presented through a coherent user interface.

Building on its core semantic processing, LexaQuery features ***enhanced natural language understanding*** to significantly improve its ability to handle complex and context-dependent queries in educational conversational retrieval.

LexaQuery maintains conversational state via a ***multi-level context model***, enabling accurate resolution of ambiguous or implicit entity references across sessions. It uses discourse analysis for pronoun resolution (based on recency and semantic compatibility) and automatically identifies implicit entities from user authentication context (e.g., enrolled courses). Domain-specific heuristics and entity salience scores further disambiguate terms, prioritizing important referents. This capability significantly improves usability and user satisfaction by enabling natural, conversational interactions.

LexaQuery accurately interprets complex negative expressions via ***sophisticated negation handling***, mapping explicit, implicit, and nested negations to SQL constructs using logical scope analysis and semantic patterns. These advanced capabilities ensure LexaQuery accurately interprets and executes queries involving negative conditions, exclusions, and exceptions, which are often challenging for other natural language database interfaces.

LexaQuery interprets ***natural language temporal references***, converting relative and absolute time expressions into precise database query constraints. It maps terms to exact date ranges using the current date and integrates with the academic calendar (semesters, terms, institutional patterns). By automatically translating these into precise date ranges and adjusting precision based on context, the system ensures accurate responses to time-dependent queries without requiring exact date input, significantly boosting usability.

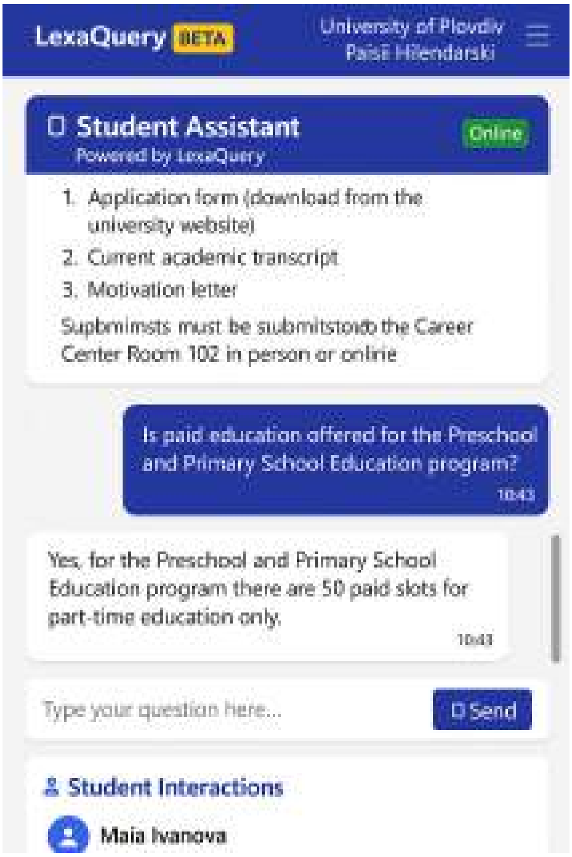
LexaQuery offers a robust natural language understanding framework for education, adeptly handling complex queries via integrated contextual entity resolution, advanced negation, and temporal expression resolution, all without requiring resource-intensive LLMs.

Security is ensured through a three-level (Student, Faculty, Administrator) role-based access control (RBAC) system. Students can view their academic records (grades, enrollments, transcripts), course schedules, exam dates, financial information (tuition, payments), and available courses for enrollment. Faculty members can access course rosters and student performance for their assigned courses, department-wide course information, and personal schedules and office hours. Administrators have comprehensive access to aggregated institutional statistics (anonymized), system-wide course and enrollment data, administrative reports and analytics. This level provides full access, with integrated audit logging for all operations. All database queries include automatic user context injection to enforce these security boundaries. Authentication integration with the university's existing Single Sign-On (SSO) system ensures seamless and secure user identification.

5. Performance evaluation

LexaQuery provides students with an intuitive chat interface for accessing university information (see Fig. 3). Students can effortlessly type natural language questions and receive immediate answers regarding their academic records, courses, exam schedules, and administrative deadlines. The system improves readability by using structured data visualization, displaying information like grades in formatted tables. All interactions are presented in a familiar messaging format, with a typing indicator that provides visual feedback while the system processes queries. LexaQuery's computational linguistics approach lets students phrase questions naturally, eliminating the need for specific commands and making university information highly accessible.

LexaQuery's performance was evaluated at the University of Plovdiv “Paisii Hilendarski” across three key areas: response time, accuracy, and resource utilization, using a dataset of over 300 real student queries, collected in Bulgarian over two months.



1. Application form (download from the university website)
2. Current academic transcript
3. Motivation letter

Supbmimsts must be submitstoto the Career Center Room 102 in person or online

Is paid education offered for the Preschool and Primary School Education program?
10:43

Yes, for the Preschool and Primary School Education program there are 50 paid slots for part-time education only.
10:43

Figure 3. GUI of LexaQuery

For comparative analysis, LexaQuery's performance was benchmarked against GPT-3.5-turbo, accessed via OpenAI's API. To ensure a fair comparison while strictly adhering to data privacy regulations (GDPR compliance), we implemented the following methodology:

– The GPT-3.5-turbo model was provided with an anonymized database schema and sample queries, critically without any actual student data;

– Real student queries were meticulously anonymized by replacing personal identifiers with generic placeholders (e.g., “[STUDENT_NAME]'s grades” became “the current user's grades”);
– A carefully crafted system prompt instructed GPT-3.5-turbo to function specifically as a university information assistant, simulating access to the described database structure;

- – Both LexaQuery and the GPT-3.5-turbo instance received identical anonymized queries to ensure an equitable performance comparison.

This approach enabled a robust performance comparison without transmitting any actual student data to external APIs, fully maintaining GDPR compliance throughout the evaluation process.

A comparative analysis of LexaQuery's performance was conducted against a cloud-hosted LLM (GPT-based API). To ensure methodological rigor and fair comparison, the LLM-based alternative received an identical database schema and university context via REST API calls. Response time, quantified from query submission to answer display, revealed a significant disparity (see Fig. 4).

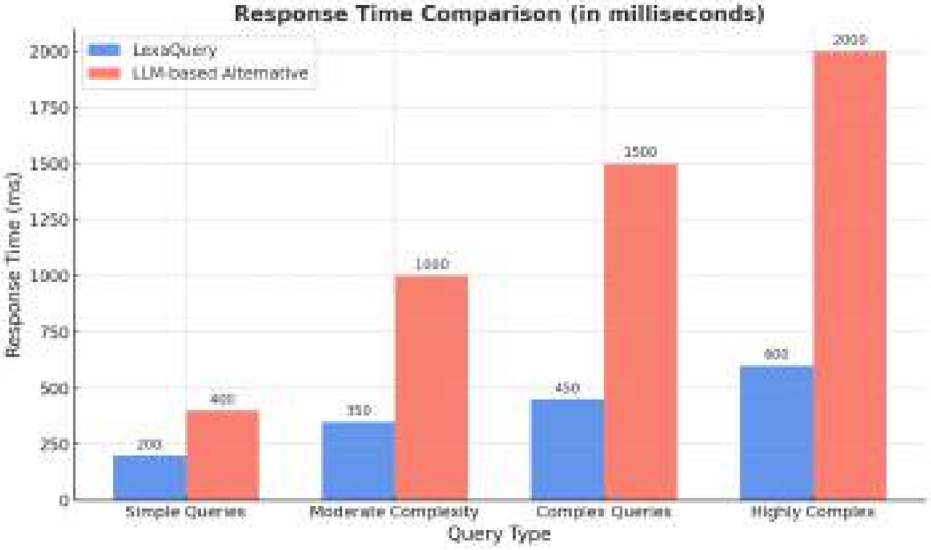


Figure 4. Response Time Comparison (in milliseconds)

LexaQuery exhibited an average response time of 178ms, in contrast to the LLM-based alternative's 1240ms. This represents an 85.6% reduction in response time, a factor of considerable importance for interactive applications. For a granular assessment of system capabilities, queries were systematically categorized into four complexity levels based on predefined criteria:

– Simple Queries (1 entity, 1 attribute, no conditions), e.g. "What is my student ID?";

– Moderate Queries (1-2 entities, 2-3 attributes, simple conditions), e.g. "When is my next exam?";

– Complex Queries (2 – 3 entities, multiple attributes, compound conditions), e.g. "Show me all courses with grades below 4.00 where I can retake exams";

– Very Complex Queries (3+ entities, aggregations, nested conditions), e.g. "What is the average grade of students in my program who took the same courses as me?".

Table 1. Accuracy evaluation results (GPT-3.5-turbo comparison with anonymized data)

Query Category	**Query Understanding (%)**		**Response Correctness (%)**	
	Lexa Query	***LLM-based Alternative***	***Lexa Query***	***LLM-based Alternative***
Academic Records	93.2	95.7	**97.5**	91.3
– Grade Inquiries	94.8	96.2	**98.9**	92.4
– Course Enrollment	91.5	95.1	**96.2**	90.2
Administrative Queries	85.6	**92.3**	**94.8**	86.7
– Document Requests	82.4	**90.9**	**93.7**	84.5
– Payment Information	88.7	**93.8**	**95.9**	88.9
Faculty Information	89.3	**93.7**	**94.2**	89.6
– Department Contacts	87.4	**92.1**	**93.1**	88.5
– Faculty Office Hours	91.2	**95.3**	**95.3**	90.7
Schedule Queries	84.8	**88.9**	**90.3**	87.2
– Class Timetables	86.5	**89.2**	**92.8**	89.5
– Exam Schedules	83.1	**88.6**	**87.8**	84.9
Complex Multi-part Queries	76.5	**89.1**	80.2	**88.4**
Overall Average	**87.3**	**91.8**	**92.1**	**89.5**

Response times were measured from query submission to complete answer delivery. For GPT-3.5-turbo, these times include network latency and API processing delays, whereas local LLM performance represents Llama-2-7B-Chat with GPU acceleration. Accuracy was assessed using two distinct metrics: query understanding and response correctness. Each response was manually evaluated by a panel of three evaluators.

For comparative analysis, LexaQuery was benchmarked against a cloud-hosted LLM (GPT-3.5-turbo via OpenAI's API) and a locally deployed LLM (Llama-2-7B-Chat with GGML quantization). To ensure fair comparison and strict data privacy, all GPT-3.5-turbo evaluations used only anonymized queries and database schema and no personal student data was transmitted. Similarly, local LLM testing used identical query sets with privacy-preserving preprocessing on consistent hardware configurations. Measured from query submission to complete answer delivery, LexaQuery averaged 178ms, a significant 85.6% reduction compared to the GPT-3.5-turbo's 1240ms (which included network latency and API processing). A panel of three university evaluators manually assessed LexaQuery's accuracy. While LLMs demonstrated a slight edge in general query understanding, LexaQuery delivered more accurate factual responses within its specific university domain, likely due to its direct database integration. Tested on an Intel Core i7-12700K (3.6GHz base, 5.0GHz boost) with 32GB DDR4 RAM and NVIDIA RTX 4070 GPU, LexaQuery proved highly efficient, averaging 12.3% CPU and 245MB RAM during query processing. In contrast, the local Llama-2-7B-Chat required substantially more resources: 47.8% CPU, 3.8GB RAM, and an additional 6.2GB VRAM usage on the GPU. This difference stems from LexaQuery's lightweight rule-based processing versus the higher computational demands of the transformer architecture, even with quantization optimizations.

Statistical significance testing using paired t-tests showed significant differences in response time ($p < 0.001$) and resource utilization ($p < 0.001$) favoring LexaQuery. Accuracy differences varied by query category, with LexaQuery showing significant advantages in factual correctness ($p < 0.05$) while GPT-3.5-turbo demonstrated superior query understanding for administrative queries ($p < 0.05$). Inter-rater reliability among the three evaluators achieved Cohen's kappa = 0.78 for query understanding assessment and ϰ = 0.82 for response correctness, indicating substantial agreement.

The results highlight some advantages of the proposed approach. LexaQuery's significantly faster response times create a more fluid and interactive user experience. It ensures strong data consistency by directly generating SQL queries from the university's existing databases. Unlike “black box” LLMs, LexaQuery's rule-based approach offers clear traceability, simplifying troubleshooting. Its lightweight design allows deployment on standard university infrastructure, eliminating the need for specialized hardware or external APIs. By processing all data locally and avoiding external API calls, LexaQuery enhances privacy for student information.

6. Conclusions

The result from pilot testing at the University of Plovdiv Paisii Hilendarski proved LexaQuery's significant advantages in response time, resource utilization, and integration with existing university systems. While LLMs continue to evolve, our work shows that for domain-specific applications with well-structured data, hybrid approaches that leverage traditional computational linguistics and database techniques can provide practical and efficient solutions that balance performance with resource constraints. The LexaQuery system represents a viable alternative for HEIs seeking to improve information accessibility without the computational overhead and complexity of deploying large neural network models.

Despite its demonstrated effectiveness in the university domain, LexaQuery's current implementation has several limitations. While efficient, the rule-based NLP is less adaptable than LLMs to unexpected or unpatterned query formulations. This means queries deviating significantly from trained patterns may necessitate manual rule updates. Moreover, adapting LexaQuery to a new HEI or domain requires substantial configuration effort, including lexicon updates, database schema mapping, and rule refinement, thus limiting rapid deployment across different institutions. LexaQuery supports Bulgarian queries through a translation layer, but its core parsing is optimized for English. Although evaluations confirm the “translation-first” method maintains accuracy, the potential for semantic nuances in Bulgarian academic terminology to be lost in translation remains. Direct native language processing could offer further performance enhancements. In addition, the system's rule-based architecture is challenged by extremely complex or ambiguous queries that demand extensive reasoning or contextual inference beyond its defined patterns. LexaQuery requires manual updates to accommodate new query patterns, changes in the database schema and institutional terminology.

Based on evaluation and user feedback, future development of LexaQuery will focus on enhancing its linguistic flexibility, learning capabilities, and integration with broader university information sources. We plan to expand linguistic patterns and the lexicon to support a wider range of query formulations and directly incorporate additional languages. Furthermore, a feedback mechanism will enable the system to refine its rules based on user interactions, alongside improving its ability to maintain context across multi-turn conversations for more natural interactions. Future work also includes semantic query optimization to boost SQL performance for complex inquiries and expanding the web scraping component to include academic publications and research findings, offering a more comprehensive view of university information.

Acknowledgements

This paper is financed by the European Union-NextGenerationEU, through the National Recovery and Resilience Plan of the Republic of Bulgaria, project № BG-RRP-2.004-0001-C01.

NOTES

1. A. Soni, Importance of AI Chatbot in the Education Industry, https://www.auxanoglobalservices.com/importance-of-ai-chatbot-in-the-education-industry/#gref
2. M. Rademaker, College of Charleston to debut AI Chatbot for student services,

https://www.live5news.com/2023/08/17/college-charleston-debut-ai-chatbot-student-services/

3. Brookdale Community College, https://www.brookdalecc.edu
4. UB Custom Publishing, https://universitybusiness.com/chatbot-engages-students-as-it-grows-knowledge-base-over-time
5. Lancater University, https://www.lancaster.ac.uk/news/lancaster-university-launch-pioneering-chatbot-companion-for-students
6. Empire State University, https://www.sunyempire.edu/
7. Salesforce.org, https://sponsored.chronicle.com/the-growing-potential-of-chatbots-in-higher-education/index.html
8. Virginia Tech, https://finaid.vt.edu
9. A. Perry, Students make new friend in Lucy the chatbot, https://www.canberra.edu.au/about-uc/media/newsroom/2018/ february/students-make-new-friend-in-lucy-the-chatbot
10. Mainstay, https://mainstay.com/case-study/community-college-boosts-student-enrollment-and-engagement-with-ai-chatbot/

REFERENCES

Abdul-Kader, S., & Woods, J. (2015). Survey on chatbot design techniques in speech conversation systems, *Int. Journal of Advanced Computer Science and Applications*, *6*(7), 72 – 80.

Annuš, N. (2023). Chatbots in Education: The impact of Artificial Intelligence based ChatGPT on Teachers and Students, *Int. Journal of Advanced Natural Sciences and Engineering Researches*, *7*(4), 366 – 370.

Augello, A., et al. (2011). A semantic layer on semi-structured data sources for intuitive chatbots, In *2011 IEEE Int. Conference on Complex, Intelligent, and Software Intensive Systems*, 477 – 482.

Cavalcanti, A., et al. (2021). Automatic feedback in online learning environments: A systematic literature review, *Computers and Education: Artificial Intelligenc*e, *2,* 100027.

Clarizia, F., et al. (2018). Chatbot: An education support system for students, *Int. Symposium on Cyberspace Safety and Security*, 291 – 302.

Cunningham-Nelson, S., et al. (2019). A review of chatbots in education: practical steps forward. In: *30th annual conference for the australasian association for engineering education: educators becoming agents of change: innovate, integrate, motivate*, 299 – 306.

Das, B., et al. (2025). Security and privacy challenges of large language models: A survey, *ACM Computing Surveys*, *57*(6), 1 – 39.

Essel, H., et al. (2022). The impact of a virtual teaching assistant (chatbot) on students' learning in Ghanaian higher education, *Int. Journal of Educational Technology in Higher Education*, *19*(1), 1 – 19.

Følstad, A., & Brandtzaeg, P. (2017). Chatbots and the new world of HCI, *Interactions*, *24*(4), 38 – 42.

Hew, K., et al. (2023). Using chatbots to support student goal setting and social presence in fully online activities: learner engagement and perceptions, *Journal of Computing in Higher Education*, *3*(51), 40 – 68.

Hien, H., et al. (2018). Intelligent assistants in higher education environments: The FIT-EBot, a chatbot for administrative and learning support. In *Proceedings of the 9th Int. Symposium on ICT*, 69 – 76.

Hiremath, G., et al. (2020). Chatbot for education system, *Int. Journal of Advance Research, Ideas and Innovations in Technology*, 4*(*3), 37 – 43.

Kooli, C. (2023). Chatbots in education and research: A critical examination of ethical implications and solutions, *Sustainability*, *15*(7), 5614.

Kowalski, S. et al. (2019). Two-way communication with chatbots: An educational perspective, In *2019 IEEE Global Engineering Education Conference*, 1497 – 1506.

Meyer, K., et al. (2023). Let's Chat: Leveraging Chatbot Outreach for Improved Course Performance, *EdWorkingPaper*, 22 – 564.

Molnár, G., & Szüts, Z. (2018). The role of chatbots in formal education, In 2018 IEEE 16th Int. Symposium on Intelligent Systems and Informatics, 197 – 202.

Ni, J., et al. (2023). Recent advances in deep learning-based dialogue systems: A systematic survey, *Artificial intelligence review*, *56*(4), 3055 – 3155.

Patel, N., et al. (2019). AI and web-based human-like interactive university chatbot (UNIBOT), In: *2019 3rd Int. conference on electronics, communication and aerospace technology*, 148 – 150.

Pérez, J., et al., (2020). Rediscovering the use of chatbots in education: A systematic literature review, *Computer Applications in Engineering Education*, *28*(6), 1549 – 1565.

Popescu, A., et al. (2003). Towards a theory of natural language interfaces to databases, In *Proc. of the 8th Int. conference on Intelligent user interfaces*, 149 – 157.

Rocio, V., & Wesley, A. (2020). Building a chatbot for student support, *Revista de Ciências da Computação*, *15.*

Sáiz-Manzanares, M., et al., (2023). Perceived satisfaction of university students with the use of chatbots as a tool for self-regulated learning, *Heliyon*, *9*(1).

Salas-Pilco, S., & Yang, Y. (2022). Artifcial intelligence applications in Latin American higher education: A systematic review, *Int. Journal of Educational Technology in Higher Education,* 19(1), 21.

Sandu, N., & Gide, E. (2019). Adoption of AI-Chatbots to enhance student learning experience in higher education in India, *In: 2019 18th Int. Conference on Information Technology Based Higher Education and Training*, 1 – 5.

Santana, R., et al. (2021). A Chatbot to Support Basic Students Questions, In LALA, *CEUR Workshop Proceedings*, 58 – 67.

system for database information retrieval, *Int. Journal of Computer Science and Network Security*, *20*(3), 189 – 198.

Sinha, S., et al. (2020). An educational Chatbot for answering queries, *Advances in intelligent systems and computing*, 55 – 60.

Tsivitanidou, O., & Ioannou, A., (2021). Envisioned pedagogical uses of chatbots in higher education and perceived benefits and challenges. In: *Inter. Conference on Human-Computer Interaction*, 230 – 250.

Vázquez-Cano, E., et al. (2021). Chatbot to improve learning punctuation in Spanish and to enhance open and flexible learning environments, *Int. Journal of Educational Technology in Higher Education*, *18*, 1 – 20.

Winkler, R., & Söllner, M., (2018). Unleashing the potential of chatbots in education: A state-of-the-art analysis, In *Academy of Management Annual Meeting (AOM)*, 1 – 40.

Yang, C. et al., (2021). Recent advances in intelligent source code generation: A survey on natural language-based studies, *Entropy*, *23*(9), 1174.

Zhang, K., & Aslan, A., (2021). AI technologies for education: Recent research & future directions, *Computers and Education: Artificial Intelligence*, *2*, 100025.

✉ **Dr. George Pashev, Assist. Prof.**
University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria
E-mail: georgepashev@uni-plovdiv.bg

✉ **Dr. Silvia Gaftandzhieva, Assoc. Prof.**
ORCID ID: 0000-0002-0569-9776
University of Plovdiv Paisii Hilendarski, Plovdiv, Bulgaria
E-mail: sissiy88@uni-plovdiv.bg