



<https://doi.org/10.15407/scine21.05.126>

IATSIUTA, V. O. (<https://orcid.org/0009-0001-8610-2862>),

and KOBETS, V. M. (<https://orcid.org/0000-0002-4386-4103>)

Kherson State University,

27, Universitetska St., Kherson, 73003, Ukraine,

+380 96 310 2636, office@ksu.ks.ua

TRANSFORMING BUSINESS COMMUNICATION WITH SOLUTIONS BASED ON ARTIFICIAL INTELLIGENCE TECHNOLOGIES WITH SUPPORT FOR NATURAL LANGUAGE PROCESSING

Introduction. Information has become one of the most critical resources in today's dynamic world, with data collection and analysis technologies evolving rapidly. Every organization requires fast, accurate, and secure data analysis to respond to queries and support informed decision-making.

Problem Statement. Artificial intelligence (AI) has become an integral part of modern life, and data processing remains at its core — from understanding a company's current state to performing complex risk assessments and forecasting decision outcomes.

Purpose. To develop a system that enhances evaluation and decision-making processes by providing users with integrated, user-friendly access to relevant datasets.

Materials and Methods. Various system architectures have been examined, with their advantages and limitations described. A prototype system for processing business queries using ChatGPT has been developed. The main input types and examples of different request scenarios have been presented, along with an overview of potential issues and proposed solutions.

Results. The system has been validated, and strategies for improving data processing quality have been proposed. The study has aimed to streamline the request lifecycle — from initial formulation to response delivery — while ensuring compliance with corporate policies and data security standards.

Conclusions. The findings have demonstrated the effectiveness of artificial intelligence in automating business query processing — specifically in transforming natural language inputs into SQL queries and generating accurate responses. This opens new opportunities for automating and optimizing enterprise processes, reducing the workload on IT departments, and improving data accessibility for non-technical users. In practical terms, the results can be applied by organizations to develop intelligent analytics systems, interactive dashboards, and database chatbots, as well as to build next-generation business intelligence interfaces based on natural language.

Keywords: business request, artificial intelligence, NLP, software architecture, data-processing.

Citation: Iatsiuta, V. O., and Kobets, V. M. (2025). Transforming Business Communication with Solutions Based on Artificial Intelligence Technologies with Support for Natural Language Processing. *Sci. innov.*, 21(5), 126–143. <https://doi.org/10.15407/scine21.05.126>

© Publisher PH “Akademperiodyka” of the NAS of Ukraine, 2025. This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

According to Gartner's data, from 2023 the number of companies on the market, which are using AI, is around 35%, and 42% of companies are exploring AI for its implementation in the future. The same source projects that the global AI market will grow to USD 266.92 billion by 2027. According to Gitnux market data report 54% of business executives say AI solutions implemented in their businesses have increased productivity. Businesses that refuse to use AI risk losing out to rivals and suffering significant financial losses. According to the results of a McKinsey poll conducted in 2021, the use of artificial intelligence (AI) in businesses is still expanding gradually. Out of those surveyed, 56% said they utilize AI in at least one function, up from 50% in 2020. According to a McKinsey survey, businesses in developing economies such as China, the Middle East, and North Africa have had the biggest surge in AI usage. Additionally, the McKinsey & Company analysis indicates that by 2030, artificial intelligence might boost the world economy by USD 13 trillion. In an MIT Technology Review survey, 81% of executives said they thought AI would give their company a significant competitive edge.

The use of AI during the first year of the COVID-19 pandemic resulted in a more than 20% decrease in the number of employees handling business functions while revenue climbed by more than 10% [1]. Companies shall comprehend the precise needs that the AI component meet before putting it into practice. These requirements include the kind of AI solution (outsourcing or in-house) and the availability of resources, data, and skills in the development and administration of such systems.

For example, government datasets are now widely available, but their use is hindered by a lack of programming skills and data management knowledge among stakeholders. The majority of data are available in unprocessed forms, which poses a technological challenge to their use or accessibility.

The rapid growth of various types of unstructured textual data, including news articles, press releases, analyst reports, annual reports, corporate

documents, blogs, social media posts, presentations, publications etc., creates the potential for a deeper understanding of the rapidly evolving business environment [2]. The purpose of the research is to create a system that improves assessment and decision-making processes by giving users integrated, user-friendly access to a collection of data.

Drawing from relevant literature, business request analysis findings, and our own extensive experience spanning over 15 years across various IT domains, we aim to bridge the gap between AI implementation and business request fulfilment by building a new type of program architecture and contrasting it with current solutions, highlighting both their advantages and disadvantages. We have developed a program prototype based on the findings and assessed its capacity to carry out the given tasks in a more efficient manner.

The remainder of the paper is structured as follows: Section 2 presents a literature review on the advantages and limitations of autonomous AI-based computations, the architecture of conversational agents, and AI as a service. Section 3 describes the research methodology. Section 4 outlines the experiment. Section 5 provides a discussion of the research findings. Finally, Section 6 contains the conclusions.

In the modern conditions of information technology development and digital transformation of business processes, there arises a need to optimize the interaction between a company and its clients, partners, or employees to reduce request processing time and improve the efficiency of the results. One effective solution in this area is the use of artificial intelligence (AI) supported by natural language processing (NLP) technologies to automate the execution of business requests.

The problem lies in the fact that traditional systems and mechanisms for processing and fulfilling business requests often require significant human resources and time, which substantially affects a company's operational costs and reduces its competitiveness compared to organizations that automate their business processes. To solve this issue, it is necessary to develop a solution that auto-

mates request processing, reducing the workload on employees while ensuring high accuracy and efficiency in handling these requests. AI, capable of understanding and interpreting natural language, can assist in this task by enabling a company to process and fulfill requests with minimal human intervention.

The research aims to develop and implement a system that uses NLP models for processing business requests, such as information requests, task execution, or decision-making. This system should automatically analyze requests, generate correct responses considering the context, ensuring high accuracy in data processing, and fast response times, without violating security policies. Additionally, the system should easily integrate into the company's existing business processes, reducing the burden on employees and allowing them to focus on more complex and creative tasks.

The main tasks of the paper are as follows:

- ◆ Analyze existing NLP technologies for integration into business processes.
- ◆ Develop the architecture for an AI system supported by NLP for processing business requests.
- ◆ Evaluate the efficiency of the developed architecture.
- ◆ Assess the security risks associated with the use of this system.
- ◆ Evaluate possible issues during system deployment and propose ways to resolve them.

The task is not only to automate simple processes but also to integrate a flexible solution that can learn and adapt to new types of requests and changes in the business environment.

The readiness assessment structure for AI considers four key dimensions: technology, activities, boundaries, and goals [1]. Before starting an AI project, it is essential to identify the users, their tasks, their potential benefits from using the system. At the early stage, it is advisable to determine the capabilities and limitations of the system.

Often, there is no single source of truth to answer queries, and periodic knowledge checks are lacking, leading to inconsistent information and facts presented to clients. Manual actions in searching

and formatting documents can be replaced by automatic generation of documents through the transformation of existing knowledge using AI methods [3]. Transforming traditional processes and services into electronic processes, such as electronic tenders, auctions and searches, allows addressing a specific purpose using data obtained from its environment, utilizing the AI's learning ability [1].

Advantages of digital transformations contain i) elimination of bureaucracy and acceleration of communication between involved parties; (ii) remote access to the system; (iii) cost reduction is a primary reason for the significant potential growth of AI [4]. Automation of document/report template development occurs through the study of model-based methods combined with AI methods. The model includes a domain-independent document system; document transformation in the model; a decision recommender based on the model to provide recommendations for client queries; transformation of the model into text for automatic document creation on request [3].

The system takes input documents in natural language, client queries, context, and automatically generates a document in response. The context of the request is considered from the client's perspective, including geography, proposals etc. (Fig. 1).

Autonomous computing explores how systems can achieve user-defined "control" outcomes independently, without human operator intervention. The integration of AI/ML to achieve such autonomous and self-directed systems can range from full automation to human-assisted automation in the loop [5].

Autonomous computing, often referred to as self-adaptive systems, envisions those systems can achieve the desired behavior autonomously. The system automatically updates missing or outdated components based on error messages generated by the monitoring system. The autonomous approach to operation is employed by IBM (Fig. 2). Manual administration and adaptation of such systems are complex and expensive. In the future, self-adaptive systems will merge with AI/ML.

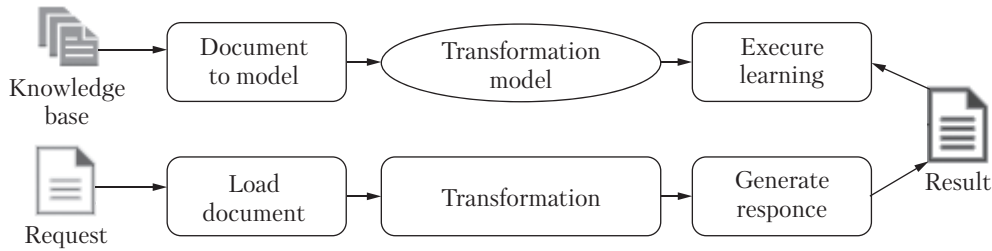


Fig. 1. Generalized approach to automating proposal development [3]

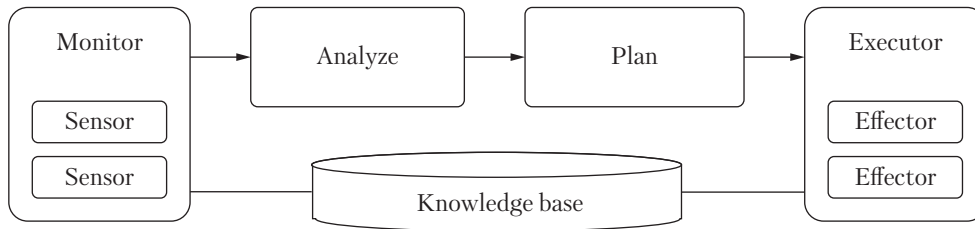


Fig. 2. MAPE-K (monitor-analyze-plan and execute) loop for autonomic computing [5]

Decision-making in business ecosystems relies on Decision Support Systems (DSS) as interactive computer information systems (Table 1) [4].

Common hybrid Decision Support Systems (DSS) include data-driven, document-driven, knowledge-driven, and web-oriented systems. DSS optimizes access to various sources of information and allows searching in structured and unstructured data to gather in-depth insights.

Requirements for AI systems should consider the quality and availability of data, how to verify them, and which data to select for training [6]. Generating large volumes of digital data enables the development of innovative IT applications that can enhance processes in both public and private institutions. Customer comments, social media posts, and written documents are examples of unstructured data that can be acquired

Table 1. Decision Support Systems Classification

DSS type	Description
Model-driven	Based on simple quantitative models, for which large databases are not required, as they operate with a limited amount of information and parameters provided to decision-makers
Data-driven	It provides access to internal corporate data for manipulation. The development of data warehouses as subject-oriented, integrated, time-variant, non-volatile sets of data, as well as On-Line Analytical Processing (OLAP). Business Intelligence (BI) encompasses a broad category of programs, technologies, and processes for obtaining, storing, accessing, and analyzing data
Group Communications-driven	It explores network and communication technologies, including group software, video conferencing, and computer dashboards
Document-driven	It facilitates document search and analysis, relying on a large database of documents (scanned documents, hypertext documents, images, sounds, video). The search system is a primary decision support tool
Knowledge-driven	Knowledge is based on computerized systems that can store and retrieve knowledge encoded in the form of probabilities, rules, and relationships using data mining and AI technologies
Web-based	All of the above can be implemented using web technology, accessible through a web browser

using Natural Language Processing (NLP) algorithms [7].

Business analytics tools assist companies in quickly generating information, enabling managers to improve efficiency. The primary goal of BI software is to help users quickly find and analyze data critical for making informed decisions in real-time. By combining structured and unstructured data from various sources, business analytics (BI) helps companies make more informed decisions [7]. Companies use BI tools for reporting and analysis, performance management, forecasting, and decision-making.

Most empirical research on engineering requirements for AI systems with a focus on RE4AI is concentrated on data management requirements. Engineering software with an AI component includes additional configuration aspects, such as data collection, selecting the appropriate algorithm (e.g., ML or NLP), and training the model based on desired input/output, with less emphasis on writing the source code. One of the challenges in writing specifications for AI software requirements is the inconsistency of input and output templates [6]. The inclusion of AI components in software systems has led to new requirements: data, ethics, ability to explain, and trust. Some non-functional requirements (NFR) for ML systems are modified to include transparency, trust, confidentiality, security, reliability, and protection [6]. As AI and ML integrate into software systems, a new role for data processing scientists emerges in this process.

Some NFRs, such as compatibility and modularity, are not as crucial in ML systems as in traditional software systems, while other requirements that were previously overlooked, such as fairness and transparency, acquire greater value. Requirements for AI ability to explain may conflict with other requirements, such as security, cost, and accuracy. Compromises need to be calculated to determine the priority of different software requirements based on AI. All these issues should be addressed in the early stages of designing AI-based software with components [6].

Types of AI systems are distinguished considering two dimensions [1]: reasoning-behavior and human productivity-rationality: Systems that think like humans, systems that act like humans, systems that think rationally, Systems that act rationally.

The advantages and challenges of autonomous computing based on AI are presented in Table 2.

The architecture of the Decision Support System (DSS) [4] consists of the following main modules: Data Collector (collects data from various sources); Pre-processing (extracts relevant data characteristics from previous step and stores them in a database for further analysis); Analyzer (performs specialized analysis of the data collected in the previous step using ML and NLP methods); Service tools (implements a specific use case). The Data Integration submodule addresses data heterogeneity obtained from various data sources and merges matching data wherever possible. The pre-processing module takes the data collected in the

Table 2. Advantages and Disadvantages of Autonomous Computations Based on AI

Advantages	Disadvantages
<p>Improvement of performance and quality, increased profitability of investments, and reduction of ownership and maintenance costs [1].</p> <p>Advantages of deploying AI in the cloud: (i) enhanced data management; (ii) automation: cloud-based AI helps companies transition from processing semi-structured documents to unstructured ones; (iii) cost savings: cloud computing allows enterprises to pay only for the resources they use and save money on maintaining massive data processing centers [5].</p>	<p>Integration: requires transferring all its programs and technologies to the cloud to create cloud-based AI.</p> <p>Inadequate data: large datasets with high-quality data are suitable for AI technologies, as the value of the solution directly depends on the data quality, but data is often disorganized or missing.</p> <p>Security and confidentiality issues: to prevent data leaks.</p> <p>AI requires significant investments and does not guarantee satisfactory results; there are difficulties in integrating AI into business processes.</p>

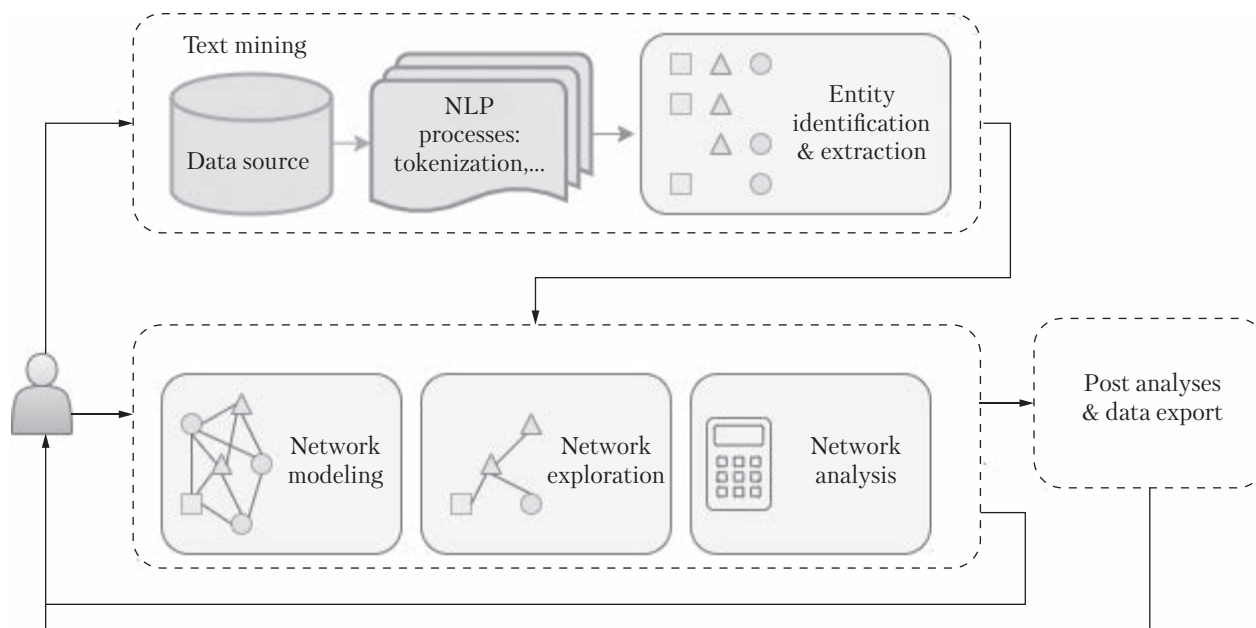


Fig. 3. Conceptual scheme of the workflow process [2]

previous stage and stores it using databases or indexes. For storing structured data, an SQL database is implemented. The obtained data in the form of CSV or JSON files are divided into several tables, each dedicated to a specific theme.

The Analyzer module analyses the data stored in memory during the pre-processing stage. This module is divided into two separate subcomponents: a data analyzer and a content analyzer. The former processes structured data (codes, dates, quantities), while the latter handles unstructured data (contracts, specifications, etc.). The core of content analysis involves NLP, which includes a search engine. Service tools are developed as a set of applications for a clearly defined set of use cases. The architecture of Conversational Agents (CA) is built as a pipeline (a chain of processes), where the agent takes the user's query as input and generates a response as output [8, 13]. Fig. 3 shows the high-level scheme of the interactive text analysis process and the capabilities of the visual analytics system.

Small and medium sized enterprises (SMEs) typically find it more cost-effective to obtain AI services from external providers. The AI as a Ser-

vice (AIaaS) business model is a new method that, like cloud services, offers AI services and solutions according to the company's needs with subscription-based or pay-as-you-go payment models, such as API-based models for ML, NLP, and image processing [1].

Advantages of this type of service for user companies [1]: No need for specialized AI skills; Proven and high-speed infrastructure; Transparent payment models based on usage; Ease of use; Scalability.

AIaaS provides AI capabilities and resources as on-demand services through the cloud, allowing organizations to use AI technologies without the need for significant initial investments in infrastructure and expertise. Cloud computing and AI reinforce each other when processing big data storage in real-time. It shifts the focus from processes and people to technology and data [5]. The Software as a Service (SaaS) paradigm is used for the successful implementation of cloud AI solutions for SMEs.

Using AIaaS, intelligent text mining explores NLP to transform unstructured text data into normalized structured data suitable for analysis or for

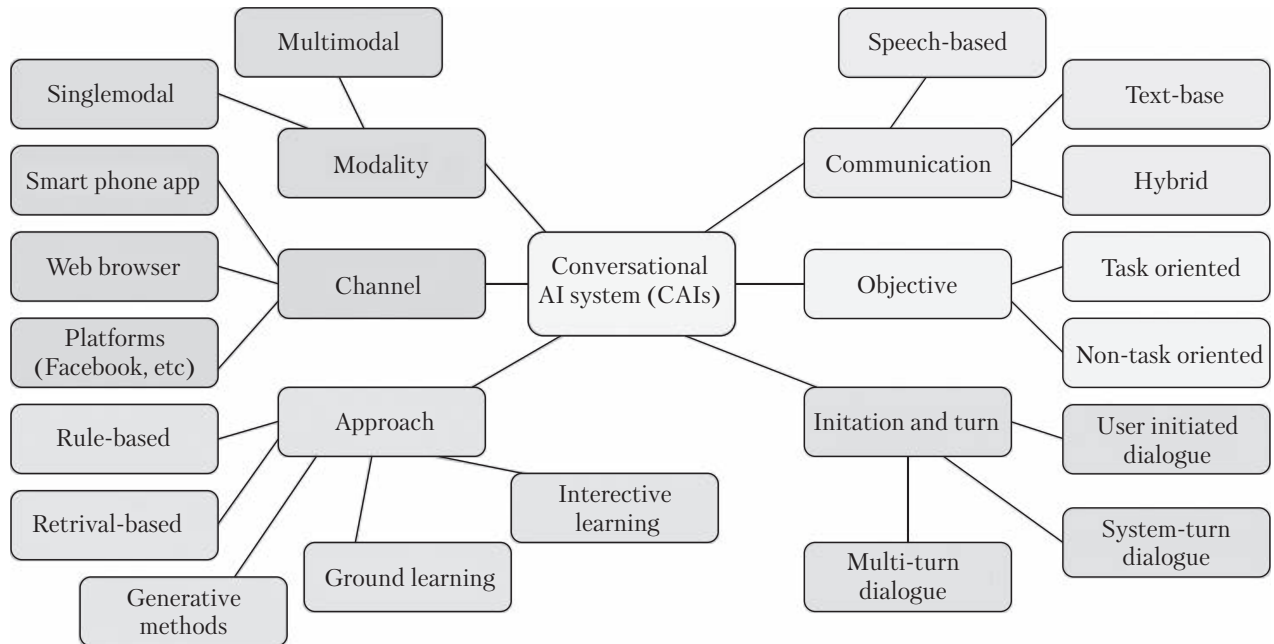


Fig. 4. Categories of Conversational AI Systems

managing ML algorithms [2]. NLP is a system that can interact with existing databases, prepare datasets for processing and analysis, perform automatic extraction of relationships between textual objects, conduct correlation tests between parts of text of different lengths, and then receive queries and return pre-defined results in a web format (short reports, evidence, reference code etc.) [4].

The creation of a Domain-Specific Language (DSL) is associated with the modelling and implementation of Conversational Agents (CA). DSL is a programming language created for use in a specific domain and to address a particular problem. DSL is defined by three main components: abstract syntax, concrete syntax, and semantics [8].

The form of interaction between humans and CAs is queries. Responses to queries based on previous or related cases can enhance the knowledge about CA participants. Industry 4.0 can automate document/report creation and streamline processes using ChatGPT's natural language capabilities. This can save time and money during the creation of coherent documents/reports without errors that meet specific requirements, utilizing ChatGPT.

Conversational Agents (CA) are programs that can communicate with users using natural language [8]. They can understand and process user queries, to take actions and/or formulate responses. Developing such applications requires expertise in software development, ML, deep learning, and NLP [8]. Currently, there is a lack of research on DSL and its connection with CAs. The use of CAs helps users access these services without installing new programs. Research [9] classified conversational AI systems into different groups based on various categories (Fig. 4).

Users can interact with Conversational AI systems using voice, text, or a combination of both. Similarly, these systems provide feedback to users through text, language, images, or a combination of these modes.

A chatbot is a combination of various technologies, such as NLP, ML, and language generation [10]. ChatGPT is an NLP tool based on AI that understands and generates text in response to provided requests. Applications for ChatGPT include chatbots, virtual assistants, automated customer support, language translation, and content creation.

The primary goal of ChatGPT is to provide human-like responses to natural language queries or prompts across different settings and topics [11]. Explainable AI (XAI) can be based on three types of analytics: performance analytics (scaled), interpretative analytics (factorial), and responsible analytics (considering existing constraints) [12].

In today's fast-paced and data-driven world, the ability to access and analyses data quickly is key to success of decision makers. Employees from different departments and positions regularly make data requests in the context of a business to support their tasks, initiatives, and decision-making processes.

To guide product development, R&D teams request data on consumer desires and market trend [14]. Risk management departments request different data in order to identify potential risks and build mitigation strategies (for example in business connected to loan management over-collateralization and potential defaults reports have a huge impact on the plan of buying/selling assets). Sales and marketing teams frequently request market data in order to better target their analytical efforts. Executives frequently go through the business information reports to help them make strategic decisions for long-run period [15]. These are just a few of the examples that support the topic's importance. Such requests are essential for increasing efficiency, making informed decisions, and boosting overall productivity. It is essential for any business to maintain a competitive advantage and keep business grow by fulfilling these requests in shortest amount of time and as accurate as possible.

Execution of requests might require significant resources, both human and financial, as well as a significant amount of time for their completion. In this paper, we have posed the research question of whether it is possible to optimize the process of executing a request from the moment of its creation to the requester's get a response, without violating security and other company policies.

Given that AI has gained unprecedented popularity in recent years, and its usage possibilities are growing every day, it is increasingly fascina-

ting more and more different fields of knowledge and can be applicable to different analytical tasks. Naturally, it is tempting to use it to solve the problems we face. Let us examine few implementations, discuss their advantages and disadvantages, and pay attention to potential security vulnerabilities and other relevant considerations. Firstly, it is necessary to understand the nature of user requests, what unites or distinguishes them, and answer the question of whether it is possible to describe a generalized process for their execution.

Drawing upon our experience (more than 15 years of work in IT on various projects from different countries and different fields of knowledge), the experience of our colleagues, and scientific researches, we can classify business requests based on the following criteria (Table 3).

Based on Table 3 requests can be divided into two types.

- ◆ The first type can be fulfilled by creating and scheduling reports, typically involving periodic static requests with moderate to high information demand [16].
- ◆ The second type requires additional research and analysis from a data analyst before providing feedback to the user. These requests are usually smaller in format, often taking the form of a question-and-answer interaction.

In this research, we have focused on the second type of request. These requests are quite common and are characterized by a constant change in requirements and the data being analyzed. Requests of this type usually do not require a large amount of information, but their complexity can be much higher than building a typical report. This is because they require a specialist who has a clear understanding of data, its locations, relationships, and possesses the knowledge of the processes involved in their proper analysis. These requests often involve navigating complex data structures, handling data security and privacy concerns, and developing custom solutions to extract [17], analyses, and present the required information. Specialized knowledge of databases, data access, and data transformation is often necessary to fulfil the-

se requests effectively. Additionally, communication between the requestor and the data specialist is crucial to ensure that the final analysis meets the specific needs and expectations.

Typically, the person making the request may not have a clear understanding of the raw data structure where proper information is located, how to process it properly. In most of the cases requester do not even have the access to data storage itself, only to the results of analysis. Let us have a look on small example. A PR manager wants to obtain statistics on bookings for hotels from specific region to understand correct approach and necessary steps need to be taken for future advertising company to have a success. To get such data manager usually needs to have information on clients (age, sex, nationality etc...), bookings, what rooms where booked, prices, type of reservation (refundable, non-refundable), breakfast type etc. Such data usually stored in databases. Only DBA, DevOps, Dev Team, QA have access to it directly. The rest teams, managers, travel agent have limited access to some data through Web interface or some corporate applications and devices. Therefore, PR manager needs to ask seller department to prepare this statistic for him/her. Depends on the data required, access level, company structure seller de-

partment might need additional assistance from other personal: travel agents, DBAs (database administrators) etc. Such request can create a long request chains, which can affect time of completion and required resources used. Providing access to database and raw data for everyone who require it (even read-only) are not going to optimize the process but rather create more chaos. Data store in database usually have complex structure and many hidden stones which ordinary person is not aware of. Therefore, with providing access company has to spent time and resources on teaching stuff of data structure etc. Moreover, as world is not standing on same place and data continuously evolving this process has to be on going. As most of these data is not used on daily based it is going to be forgotten shortly after it has been learned and we going to need to repeat the learning process again. Also providing full read access to everyone is dangerous form safety perspective, personal data; credit card information, history, home address etc. should not be accessed by any. The more persons have access to it the higher chances of data-leaking and inappropriate information usage. This can cause high reputation problems, job losses and even bankruptcy. Additionally knowing data structure is not guarantee effective usage of

Table 3. Classification of Business Requests

Criteria of business requests	Classification of queries types
1. Amount of Required Information (depends on how much information is needed)	Low Information: Requests that require minimal information or details Moderate Information: Queries that need a moderate amount of information or specifics High Information: Demands for a significant amount of detailed information
2. Response Format	Textual Response: Requests where responses are expected in plain text or written format Visual Representation: Queries that require responses in the form of charts, graphs, or images Interactive Response: Requests for responses that involve user interaction or dynamic content
3. Regularity (depends on how frequently these requests appear)	One-Time Requests: Queries that are made as singular or infrequent instances Periodic Requests: Demands that occur at regular intervals or on a schedule Continuous Updates: Requests that require ongoing, real-time updates or modifications
4. Frequency of Changes (how often adjustments or modifications are necessary)	Static Requests: Queries where information remains relatively constant over time Dynamic Requests: Demands that require frequent updates or changes due to evolving circumstances or data

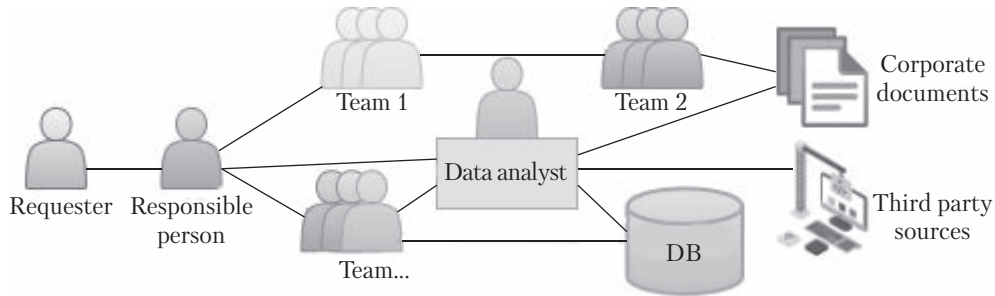


Fig. 5. Generalized scheme of second type requests

it, as it requires deeper knowledge of databases and business where people are working in. In addition, if company works with third-party data one might need to know and understand how their business works too including process involved in data analysis [18]. As we mentioned before a chain of persons and departments involved in solving the request can increase pretty fast eating more and more company resources.

Let us have a look at a small example from one of the projects we have participated in. The travel company owns a chain of hotels around the globe. On their website, their purpose varies types of rooms with different types of policies (non-refundable, with breakfast, without breakfast, with ability extend etc...). Additionally, they aggregate information from third party hotel suppliers (Hilton, Best Western, Olympia etc.) with the ability to reserve the room. In addition, they provide the ability to order transportation, book sightseeing, and a bunch of other type of the services. The request came from one of the top managers at a board meeting and sounded simple: calculate the potential losses and expenses company faces in case all trips are cancelled in one particular region due to an unexpected situation. The request was passed to one of the travel managers. Through the website, they had limited access to the number of reserved rooms with specific customer information and booking types. It started with a request to the Legal team to understand the types of cancellation policies in different hotels and who would cover the expenses in case of unexpected situations. The question arose regarding the type of unexpected situation. If we

mention a typical COVID outbreak and the hotel closes due to quarantine that is one scenario. If the COVID outbreak happens, but the hotel operates while the state closes airports that is another scenario. If we talk about natural disasters like earthquakes or floods, those are entirely different cases, not to mention military conflicts.

Upon receiving detailed information from the Legal team, the travel manager decided to create several reports. However, they lacked additional information on cancellation fees for specific room types, payment systems in different hotels, and more. Third-party hotels supported various payment systems, sometimes even offering credits and instalment plans at different interest rates. The travel manager had to send requests to the product owner, development team, data analysts, and even write direct letters to hotels. This example describes only part of the interaction chains. We do not know how deep the request went, such as when contacting the hotel, or how many people were involved in gathering data, like the Legal team, and whether they sought help from other teams.

If we look at the original; request from the desk, it leans more towards the first type since it requires a substantial amount of information for processing and could potentially be recurring in the future for other periods and locations. However, the sub-requests made by the travel manager to the Legal team, hotels, etc., mostly fall into the second type of requests.

Based on all the above, let us propose to create a generalized scheme for describing a process of fulfilling requests of the second type (Fig. 5).

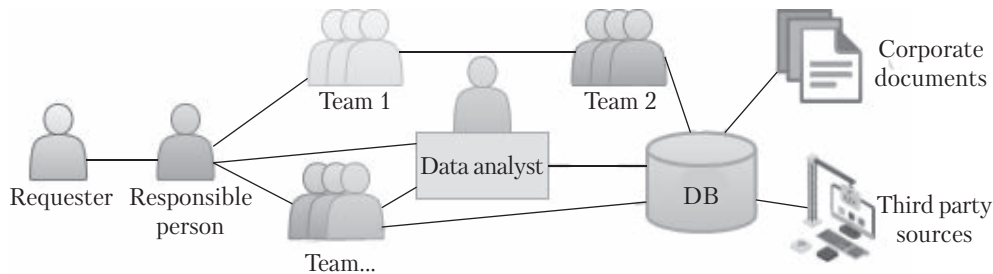


Fig. 6. Optimized scheme of second type requests

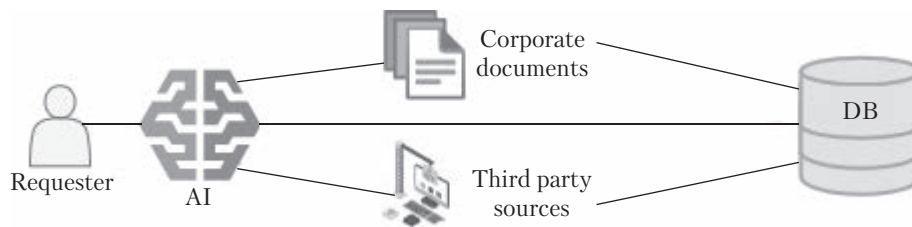


Fig. 7. AI-based architecture of requests scheme

Our next goal is to optimize the process with efficient usage of artificial intelligence. The first step could be optimizing the data and creating a unified storage space, for example databases can be used for it. Most business structures typically aim to store all necessary data in a single structure with configured replication (when using a multi-server architecture) and backup mechanisms in case of data loss. This approach allows optimizing data access mechanisms and facilitating their storage and transfer if necessary. This especially applies to third-party data (data not directly owned by the enterprise and usually sourced from external providers using Rest APIs, files on FTP servers, scraping mechanisms, etc.). The synchronization and storage processes of such data into a unified structure vary significantly, from handling massive files sent by third-party sources (or downloaded from their servers) to scheduled jobs that periodically collect data from provided APIs. In some cases, scraping mechanisms are used, where automated tests written in Selenium (or any other tool), mimic human behavior to gather information directly from websites etc. Optimized data sources scheme for requests is presented in Fig. 6.

Once we have optimized the data sources, the next question is how to effectively utilize AI to enhance this architecture. One of the most apparent options is an architecture where the user directly interacts with AI supported by NLP, specifically created for the enterprise and trained on its data. Using external data from other enterprises for training is possible, but it might affect the quality of the results (both positively and negatively) and significantly complicate the validation process of complex requests. Additionally, it is crucial to maintain continuous training AI, with all new data passing through it (Fig. 7).

The primary advantages of this approach include reliability and security (data remains on the enterprise's servers and is not transferred to external sources). However, there are some drawbacks in accuracy, dependency, significant cost. Such approach requires substantial resources such as multiple teams with deep knowledge in AI to build the architecture, buy and maintain servers, personnel to support the AI system etc. [19].

The alternative architecture involves utilizing existing Artificial Intelligence with NLP support (for example, like chat GPT) as an intermediary between the user and the data [20]. Following prior

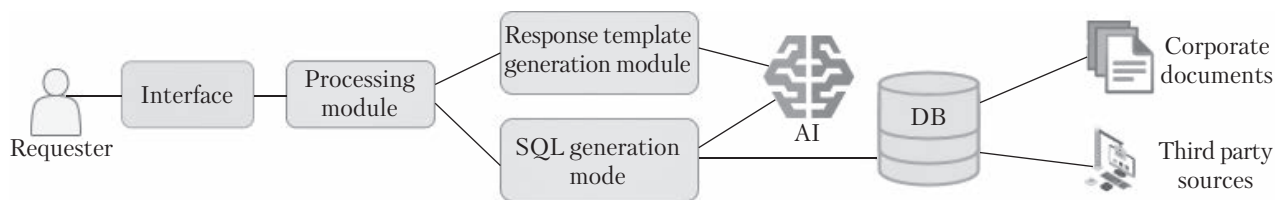


Fig. 8. AI-based architecture of requests scheme with NLP support

optimization where all necessary data has been moved to a centralized repository (such as a database), the AI is entrusted with the following functions:

- ◆ Analyzing user queries written in natural, human language.
- ◆ Formulating a query to the centralized repository (SQL, NoSQL query, etc.) based on the analyzed request.
- ◆ Crafting a response based on the obtained results and delivering it back to the user.

The resulting architecture would appear as follows (Fig. 8):

As seen in the diagram, the user sends a query to the server through some interface. Subsequently, it is forwarded to the Artificial Intelligence module, which translates it into a query for the database and sends it to the server. The server, having access to the database, executes the query. Afterward, the obtained results are sent back to the AI module for the formulation of a response to the user. In most cases, what we primarily need from the AI is not just an answer, but also rather a response template where we can insert the query results.

As this option is the most accessible, let us focus on it. As previously mentioned, the development and use of AI are progressing at an enormous pace. To demonstrate the relevance of this work, we have checked the market for products, which are using a similar or identical approach to what we have proposed. The closest product we have managed to find was a library LangChain [21]. We successfully launched it in Python (we also attempted to do it in Java, but due to errors in the library, we were unable to). Unfortunately, the results we obtained were unsatisfactory. This particular library is limited only to the MySQL server database, requires direct access to the data-

base, for every user query it sends a complete script of database creation with several lines of data from each table, which is a significant security breach. Additionally, it does not restrict the user from altering data, which could compromise their integrity. The large data sizes transferred with each query significantly impact processing time and the cost of using AI.

To continue work on this topic we have to answer few questions. First of them is how to properly and optimally organize the work between the company's servers and AI [22]. The interaction process should be flexible and not depend on the type of database. Additionally, considering that the company does not own the AI servers, limitations have to be placed on the data that can be transferred. However, for the correct construction of the query, the AI module needs to know the specific characteristics of the stored data and how to process them correctly without compromising their integrity. We also need to prevent the possibility of AI altering the data for obvious reasons.

Let us start with organizing the work between Artificial Intelligence and the company's servers. Most AI providers support an API that can be used for both commercial and general purposes. To do this, it is enough to fund the wallet and create an API token. This token should be passed in the HTTP request headers to AI REST API and is used for client authentication. Depending on the platform, a fee is deducted from the wallet (usually for a certain amount of transmitted characters). AI API usually provides response in JSON format containing generated text based on requested information and additional data like id, model, creation data-time etc. The structure of JSON itself cannot be changed, however the text in the

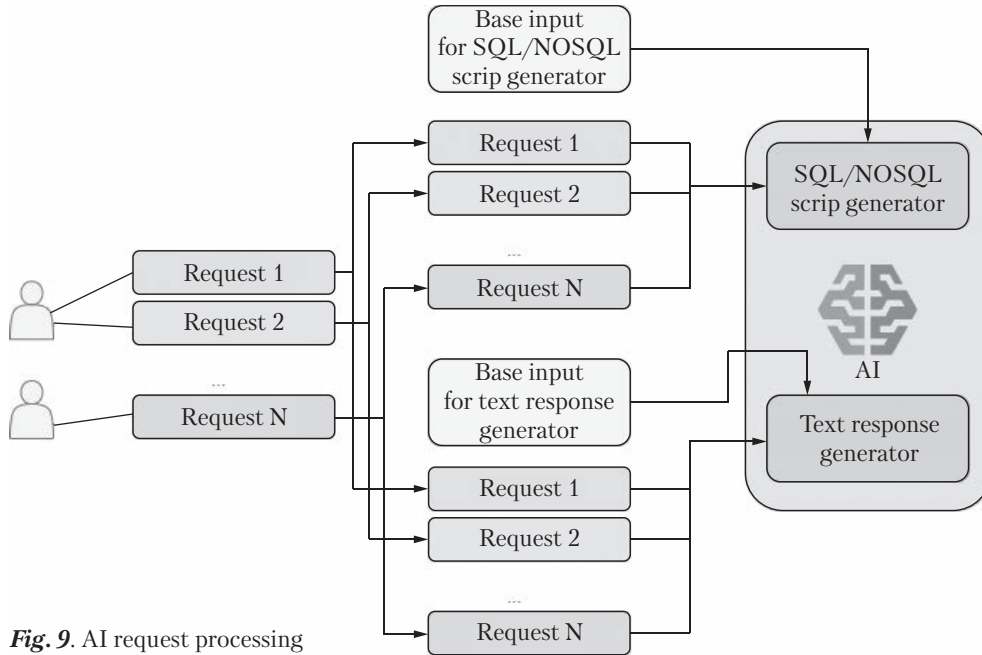


Fig. 9. AI request processing

response can have certain format, which can be parsed to get information we need and be reused for our purposes.

For our task, we need two types of response: Text response template; SQL/NOSQL script. Since AI retains user-inputted data for a certain period and uses it to improve response generation, we can create two different chat/user types (chat/user groups for a large number of users), each for a specific type of task. For each of them, a basic input needs to be established, explaining the task type. This input should be provided only at the beginning, during the chat/user generation, for their configuration. Subsequently, requests can be processed one after another without the need for additional information. However, it is important to remember that time to time, the AI may reset the settings by itself or this might happen due to a user-inputted request. In such cases, a mechanism to check the AI's responses needs to be implemented, and if response does not match the required format, a new chat/user should be created (Fig. 9).

We are going to need two templates:

- ◆ If an answer to the question cannot be found: in this case, we can use a generic response

template(s) without assistance from AI. For example:

"I'm sorry, but I don't have information on the question asked. Please try rephrasing it."

- ◆ If an answer has been found: in this case, we can use the same approach as before and create template by ourselves. For example:

"Here is the answer to your question"

If we want the answer to be more human friendly, we need the help from AI to generate a response similar to a human. To do this, we need provide AI input with explanation of what we need from it (additionally we need to use some examples). Input can look like (Provided input and follow-up questions has been tested in OpenAI ChatGPT [23]):

*You need to generate the response to questions using *** as an answer placeholder. Example:*

Question: "How many bookings were made by male clients in the period 2023-10-11 – 2024-11-11"

*Answer I need: "Around *** booking where made in 2023-10-11 – 2024-11-11 period by male clients"*

After that, we can try to ask few follow-up questions to test, for example:

- ◆ Question: *How many tours were sold last year?*

AI generated response: “*Last year, a total of *** tours were sold.*”

◆ Question: What is an average price for a triple room of Hilton hotel located in New Jersey

AI generated response: “*The average price for a triple room at the Hilton hotel in New Jersey is around ***.*”

◆ Question: What is average price for a book?

AI generated response: “*The average price for a book is typically around *** dollars.*”

◆ Question: Name and price of the last book we have sold

AI generated response: “*The last book sold was named *** and was priced at *** dollars.*” – in this case we have requested two parameters, both properly have been replaced with ***.

In some cases, AI failed to generate template:

◆ Question: *What is the time right now?*

AI generated response: “*I can’t provide real-time information, but you can check the current time on your device or by searching online.*”

The examples provided demonstrate that we need to be cautious when using templates generated by AI, as they may not always be generated correctly or be suitable for forming a response to the user. For instance, in the last example, the template might not be usable at all, while in the response about the average book price, the price is described in dollars, which might not correspond to reality and could mislead the user (for example, in the European market, the price could be in euros...). If we still want to use AI for this task, we need to extend the input and include additional criteria: what currency should be used, business idioms etc.

For SQL/NOSQL script generation the approach is similar to text response.

Let us focus on SQL-based databases as they have a wider range of usages. (The approach for NoSQL is similar). First, we need to create base input. Explaining AI what we need: we can use for example:

You need to generate only SQL queries, without any additional text, in response to the questions I ask based on a MySQL database

```
CREATE TABLE Booking (
    rental_id INT(10) NOT NULL,
    booking_date DATETIME(19) NOT NULL,
    customer_id SMALLINT UNSIGNED(5) NOT NULL,
)
```

```
CREATE TABLE Customers (
    customer_id INT(10) NOT NULL,
    first_name VARCHAR(45) NOT NULL,
    last_name VARCHAR(45) NOT NULL,
    birthdate DATETIME(19) NOT NULL,
    sex INT(10) NOT NULL -- 1 male, 2 female
)
```

In addition, let us ask follow-up questions

◆ Question: *How many bookings were made by female clients with age over 30 in the period 2023-10-11 – 2024-11-11*

AI Answer:

```
SELECT COUNT(*) AS num_bookings
FROM Booking b
JOIN Customers c ON b.customer_id = c.customer_id
WHERE c.sex = 2 -- Female
AND TIMESTAMPDIFF(YEAR, c.birthdate, CURDATE()) > 30
AND b.booking_date BETWEEN '2023-10-11'
AND '2024-11-11';
```

◆ Question: *What is the age and the name of the youngest client?*

AI Answer:

```
SELECT first_name, last_name,
TIMESTAMPDIFF(YEAR, birthdate, CURDATE())
AS age
FROM Customers
ORDER BY birthdate ASC
LIMIT 1;
```

◆ Question: *Date of oldest reservation we have*

AI Answer:

```
SELECT MAX(booking_date) AS oldest_reservation_date
FROM Booking;
```

As we can see, the generated SQL queries sufficiently fulfill the requests; the ‘join’ statements were constructed correctly even without using primary/foreign keys. However, it is important to note that to ensure the accuracy and reliability

lity of the generated SQL we need to fulfil certain criteria:

- ◆ The database type, version etc. are specified. (Different databases support different functions, types, etc.)
- ◆ Table and column names have clear and unambiguous identifiers.
- ◆ Comments are used to clarify unclear data (for example customer sex).
- ◆ Primal/Foreign keys are utilized in cases of non-obvious relationships within the database.
- ◆ For tables with big amount of data we need to have Indexes, to ensure the request are not filling due to timeout.
- ◆ Additionally, all the specificities present in the data should be added to the basic input. Here some examples:
 - table stores not only active data; it contains fields like ‘IsActive’.
 - table contains historical data using insertion and expiration dates, or versioning through history tables, or separate tables for latest and historical data.
 - column contains complex data (JSON, String separated by ‘,’ etc.)
- ◆ The base input should continuously support and updated in time, as data changes and evolve.
- ◆ Since the results could be substantial in size, it is necessary to add a limit during their load, (users should be informed in advance). This prevents overloading the database. The best way to implement this is at the SQL script calling level since these settings can be reset by the user’s query during script creation.

The next issue we want to address is ensuring security and data integrity. As we have previously mentioned, violations of company security policies can lead to extremely negative consequences.

To ensure appropriate security level, we propose the following steps:

- ◆ The database user used for processing SQL scripts should have read-only privileges to prevent unauthorized data alteration.
- ◆ In initial input data for AI, remove all columns containing sensitive data such as passwords/

cached passwords, payment data etc. from the database script.

- ◆ In initial input data for AI remove all execution procedures.
- ◆ Implement additional validation for scripts before their execution.
- ◆ Restrict the transmission of any information to AI beyond the necessary query and data required for its correct processing.
- ◆ Conduct training sessions for staff to prevent the inclusion of sensitive data in requests to AI.
- ◆ Maintain a log of all data transmitted to and received from AI, conducting regular analysis and reviews.
- ◆ When necessary to provide examples of stored data for learning purpose (for instance, when storing complex structures in one column), use dummy data (unreal information what used as a placeholder for testing and operational purposes).

Finally, yet importantly, we wanted to discuss a validation. Building a proper validation process is essential to enhance base input, improve response quality and accuracy, and decrease the number of mistakes that can occur. The process of AI response validation is complex due to the dynamic nature of language, context-dependent interpretations, and the evolving nature of AI models. It is not enough to just check and validate the results but to understand nuances of data structure and business processes inside the company.

Possible solutions that can be used independently or combined are as follows:

- ◆ Human monitoring: having human moderators to review AI-generated responses can help catch errors or inaccuracies that automated checks might miss. Unfortunately, this cannot be done in real-time, for obvious reasons. We propose to store the requests sent by the user and the responses generated by AI in a database, periodically checking them. Paying particular attention to requests for which no answers could be found or where AI was unable to generate a script, as well as requests with significant sizes.

- ◆ Testing: For manual testing, it suffices to create and run a list of test scenarios, updating it as needed (during the development of new functionality, changes to existing functions, data evolution, etc.). Implementing automated testing is more complex since it requires creating a mock version of the data and maintaining its current state by updating data, structure, tests etc.
- ◆ Gathering feedback from users who interact with the AI can provide valuable insights into the quality and accuracy of the responses.

CONCLUSIONS

In this paper, we have explored and analyzed various types of business requests and the possibilities of using AI for their processing. We have presented different system architectures, outlining the positive and negative aspects of each. As part of our experimentation, we have developed a prototype system for processing business requests using ChatGPT. We have provided basic input data, examples of processing different types of re-

quests, and described potential challenges along with proposed solutions. Validation has been highlighted as a separate component, and we have proposed ways to improve the quality of data processing.

Additionally, we have analyzed existing solutions. The analysis has been based on the processes of handling various business requests and the outcomes of such processing. We have compared the obtained data with those from the developed software prototype. The results have revealed potential security vulnerabilities, optimization challenges, and limitations in supporting multiple data sources within existing solutions. In our study, we have identified these issues and demonstrated how the proposed architecture addresses them.

The use of AI for processing business requests has proven to be a promising direction that can significantly enhance and streamline the mechanisms for handling such queries. The advantages include high processing speed, while challenges remain in areas such as validation and the creation of appropriate input data.

REFERENCES

1. Hajipour, V., Hekmat, S., Amini, M. (2023). A value-oriented Artificial Intelligence-as-a-Service business plan using integrated tools and services. *Decision Analytics Journal*, 8, 100302. <https://doi.org/10.1016/j.dajour.2023.100302>
2. Basole, R. C., Park, H., Seuss, C. D. (2024). Complex business ecosystem intelligence using AI-powered visual analytics. *Decision Support Systems*, 178, 114133. <https://doi.org/10.1016/j.dss.2023.114133>
3. Nistala, P., Rajbhaj, A., Kulkarni, V., Noronha, S., Joshi, A. (2024). An industrial experience report on model-based, AI-enabled proposal development for an RFP/RFI. *Science of Computer Programming*, 233, 103058. <https://doi.org/10.1016/j.scico.2023.103058>
4. Siciliani, L., Taccardi, V., Basile, P., Ciano, M. D., Lops, P. (2023). AI-based decision support system for public procurement. *Information Systems*, 119, 102284. <https://doi.org/10.1016/j.is.2023.102284>
5. Gill, S. S., Xu, M., Ottaviani, C., Patros, P., Bahsoon, R., Shaghghi, A.,..., Uhlig, S. (2022). AI for next generation computing: Emerging trends and future directions. *Internet of Things*, 19, 100514. <https://doi.org/10.1016/j.iot.2022.100514>
6. Ahmad, K., Abdelrazek, M., Arora, C., Bano, M., Grundy, J. (2023). Requirements engineering for artificial intelligence systems: A systematic mapping study. *Information and Software Technology*, 158, 107176. <https://doi.org/10.1016/j.infsof.2023.107176>
7. Tripathi, M. A., Madhavi, K., Kandi, V. S. P., Nassa, V. K., Mallik, B., Chakravarthi, M. K. (2023). Machine learning models for evaluating the benefits of business intelligence systems. *Journal of High Technology Management Research*, 34, 100470. <https://doi.org/10.1016/j.hitech.2023.100470>
8. Ouaddia, C., Benaddia, L., Jakimia, A. (2024). Architecture, Tools, and DSLs for Developing Conversational Agents: An Overview. *Procedia Computer Science*, 231, 293–298. <https://doi.org/10.1016/j.procs.2023.12.206>
9. Saka, A. B., Oyedele, L. O., Akanbi, L. A., Ganiyu, S. A., Chan, D. W. M., Bello, S. A. (2023). Conversational artificial intelligence in the AEC industry: A review of present status, challenges and opportunities. *Advanced Engineering Informatics*, 55, 101869. <https://doi.org/10.1016/j.aei.2022.101869>

10. Wei, R., Pardo, C. (2022). Artificial intelligence and SMEs: How can B2B SMEs leverage AI platforms to integrate AI technologies? *Industrial Marketing Management*, 107, 466–483. <https://doi.org/10.1016/j.indmarman.2022.10.008>
11. Javaida, M., Haleema, A., Singh, R. P. (2023). A study on ChatGPT for Industry 4.0: Background, potentials, challenges, and eventualities. *Journal of Economy and Technology*, 1, 127–143. <https://doi.org/10.1016/j.ject.2023.08.001>
12. De Bock, K. W., Coussement, K., De Caigny, A., Slowiński, R., Baesens, B., Boute, R. N., Choi, T., ..., Weber, R. (2023). Explainable AI for Operational Research: A defining framework, methods, applications, and a research agenda. *European Journal of Operational Research*, 317(2), 249–272. <https://doi.org/10.1016/j.ejor.2023.09.026>
13. Ouaddi, C., Benaddi, L., Khriess, I., Jakimi, A. (2023). Developing Conversational Agent Using Deep Learning Techniques. *Computer Sciences & Mathematics Forum*, 6. <https://doi.org/10.3390/cmsf2023006003>
14. Kenton, W. Business Segment Reporting Definition, Importance, Example. URL: <https://www.investopedia.com/terms/b/business-segment-reporting.asp> (Last accessed: 03.07.2024).
15. Bondar, S., Ruppert, C., Stjepandić, J. (2014). Ensuring data quality beyond change management in virtual enterprise. *International Journal of Agile Systems and Management*, 7(3–4) 304–323. <https://doi.org/10.1504/IJASM.2014.065346>
16. Iatsiuta, V., Kobets, V., Ivanov, O. (2023). Creating of a General Purpose Language for the Construction of Dynamic Reports. In: Maślankowski, J., Marcinkowski, B., Rupino da Cunha, P. (Eds.) Digital Transformation. PLAIS EuroSymposium 2023. *Lecture Notes in Business Information Processing*, vol. 495. Springer, Cham. https://doi.org/10.1007/978-3-031-43590-4_2
17. Nguyen, M.-T., Le, D. T., Le, L. (2021). Transformers-based information extraction with limited data for domain-specific business documents. *Engineering Applications of Artificial Intelligence*, 97, 104100. <https://doi.org/10.1016/j.engappai.2020.104100>
18. Kobets, V., Yatsenko, V., Mazur, A., Zubrii, M. (2020). Data analysis of personalized investment decision making using Robo-Advisers. *Science and Innovation*, 16(2), 80–93. <https://doi.org/10.15407/scine16.02.080>
19. Kobets, V., Tsiuriuta, N., Lytvynenko, V., Novikov, M., Chizhik, S. (2020). Recruitment web-service management system using competence-based approach for manufacturing enterprises. *Lecture Notes in Mechanical Engineering*, 138–148. https://doi.org/10.1007/978-3-030-22365-6_14
20. Khurana, D., Koli, A., Khatler, K., Singh, S. (2023). Natural language processing: state of the art, current trends and challenges. *Multimed Tools Appl.*, 82, 3713–3744. <https://doi.org/10.1007/s11042-022-13428-4>
21. LangChain. URL: https://python.langchain.com/docs/get_started/introduction (Last accessed: 28.06.2024).
22. Davenport, T. H., Ronanki, R. Artificial Intelligence for the Real World. URL: <https://hbr.org/2018/01/artificial-intelligence-for-the-real-world> (Last accessed: 28.06.2024).
23. OpenAI API, GPTs, Documentation and CookBook. URL: <https://chat.openai.com/g/g-I1XNbsyDK-api-docs> (Last accessed: 28.06.2024).

Received 14.07.2024

Revised 01.10.2024

Accepted 16.10.2024

В.О. Яцюта (<https://orcid.org/0009-0001-8610-2862>),

В.М. Кобець (<https://orcid.org/0000-0002-4386-4103>)

Херсонський державний університет,
вул. Університетська 27, Херсон, 73003, Україна,
+380 96 310 2636, office@ksu.ks.ua

ТРАНСФОРМАЦІЯ БІЗНЕС-КОМУНІКАЦІЇ ЗА ДОПОМОГОЮ РІШЕНЬ НА ОСНОВІ ТЕХНОЛОГІЙ ШТУЧНОГО ІНТЕЛЕКТУ З ПІДТРИМКОЮ ОБРОБКИ ПРИРОДНОЇ МОВИ

Вступ. У сучасному динамічному світі інформація є одним із найважливіших ресурсів, а технології збору та аналізу даних стрімко розвиваються. Саме тому кожна організація потребує швидкого, точного та безпечного аналізу даних, щоб відповідати на запити й робити вибір.

Проблематика. Штучний інтелект (ШІ) стає невід’ємною частиною життя, а обробка даних — його фокусом — від розуміння поточного стану справ у компанії до складних розрахунків потенційних ризиків та наслідків прийняття рішень.

Мета. Створення системи, яка покращить завдання оцінювання та процеси прийняття рішень шляхом надання користувачам інтегрованого, зручного доступу до набору даних.

Матеріали й методи. Показано різні архітектури системи, описано позитивні й негативні сторони кожної з них. Розроблено прототип системи для обробки бізнес-запитів з використанням *ChatGPT*, подано основні вхідні дані, приклади обробки різних запитів, а також описано потенційні проблеми та шляхи їх вирішення.

Результати. Проведено валідацію системи й запропоновано потенційні рішення для покращення якості обробки даних. Дослідження спрямовано на вдосконалення процесу виконання запитів від моменту створення до отримання відповіді запитувачем без порушення корпоративних політик та правил безпеки.

Висновки. Отримані результати демонструють ефективність застосування штучного інтелекту для автоматизованої обробки бізнес-запитів, зокрема їхнього перетворення в *SQL*-запити та формування релевантних відповідей, що відкриває перспективи для автоматизації та оптимізації бізнес-процесів на підприємствах, зниження навантаження на ІТ-підрозділи й підвищення доступності даних для користувачів без технічних знань. У практичному аспекті результати можуть бути використані на підприємствах і в організаціях для створення інтелектуальних систем аналітики, інтерактивних дашбордів, чат-ботів для взаємодії з базами даних, а також для побудови нових інтерфейсів бізнес-аналітики, що працюють на основі природної мови.

Ключові слова: бізнес-запит, штучний інтелект, NLP, архітектура програмного забезпечення, обробка даних.