DOI: https://doi.org/10.53052/9788367652124.01

Oleg BATRAK[1], Mykhailo GOLOVAN[2], Viktor GNATYUK[3]

OPTYMALIZACJA DZIAŁANIA SYSTEMU USŁUG MASOWYCH Z WYKORZYSTANIEM SZTUCZNEJ INTELIGENCJI

Streszczenie: W artykule omówiono nową metodę optymalizacji działania system usług masowych, używając wirtualnego asystenta stworzonego metodami sztucznej inteligencji – jako efektywne narzędzie realizujące proces obsługi użytkownika/klienta w sposób automatyczny i bardziej efektywny.

Słowa kluczowe: centrum kontaktowe, system usług masowych, sztuczna inteligencja

OPTIMIZATION OF MASS SERVICE SYSTEM OPERATION USING ARTIFICIAL INTELLIGENCE

Summary: The paper developed a new method of optimizing the operation of mass service systems, using a virtual assistant based on artificial intelligence as an effective tool for automating and improving user service processes.

Keywords: contact center, mass service system, artificial intelligence.

1. Introduction

The relevance of developing new methods for optimizing the operation of mass service systems (MSS) in today's world is important for several key reasons: improving the quality of service (more efficient mass service systems allow for improved service quality and customer satisfaction; optimization can include reducing waiting times, optimizing queues and ensuring fast and efficient service), efficient use of resources (MSS optimization allows rational use of resources, such as working time, personnel, equipment and infrastructure; this contributes to the effective use of funds and increased productivity), the use of modern technologies (modern technologies, including artificial intelligence (AI), data analysis and automation, open up new opportunities for optimization of MSS; implementation of these technologies allows to create more intelligent and efficient systems), increase business

[1] National Aviation University, Educational scientific institute of continuous education, Electronic communications and radio engineering, olegh.batrak@npp.nau.edu.ua
[2] National Aviation University, Faculty of Aeronautics, Electronics and Telecommunications, Electronic communications and radio engineering, 2199038@stud.nau.edu.ua
[3] National Aviation University, Faculty of Aeronautics, Electronics and Telecommunications, Electronic communications and radio engineering, viktor.hnatiuk@npp.nau.edu.ua

competitiveness (optimization of MSS work allows companies to increase competitiveness in the market; quick and high-quality service becomes an important competitive advantage that attracts more customers and improves their brand perception), response to modern challenges and trends (consumer requirements and technological capabilities change over time; MSS optimization allows you to respond to new challenges and adapt to changes in consumers I asked). Therefore, the development of new methods for the optimization of MSS is an urgent scientific task, as it contributes to the improvement of the quality of service, more efficient use of resources, the use of modern technologies and the increase of business competitiveness.

Modern methods of optimizing the work of MSS include a large number of technologies, models and strategies. Here are a number of scientific studies of modern methods: the author presents the general theory of queues and optimization methods in MSS [1], the publication is devoted to the modeling and analysis of service systems, including the distribution of tasks between resources [2], the article is devoted to methods of optimizing the distribution of resources in cloud computing for effective service tasks [3], the article considers the methods of queue length estimation and call permission management in networks with services with different characteristics [4], the article proposes machine learning methods for selecting web services taking into account the quality of service [5]. These scientific papers present some of the current MSS optimization techniques used in various industries such as telecommunications, cloud computing, and web services.

Considering the results of the analysis, it is worth noting that modern methods of optimization of MSS have their advantages, but there are also certain disadvantages. Disadvantages of modern MSS optimization methods: complexity of modeling, sensitivity to parameters, limitations in real conditions. Also, it is worth noting the advantages of MSS using a virtual assistant (VA) (telegram bot): automation and efficiency, improvement of service quality, constant availability and speed of response, reduction of costs, facilitation of interaction with users. Integrating VA into a MSS can help maximize the benefits of automation and improve user interaction. However, it is important to consider them as an additional tool, and not as a full replacement for the human factor and expertise.

2. Development of a method for optimizing the operation of the MSS using AI

To develop VA, it is necessary to choose tools, a platform and perform the following stages.

Stage 1. Registration of a virtual assistant.

In order to register a VA, it is necessary to choose a platform, according to studies in works [11, 12], let's choose the Telegram instant messaging system as an example. To develop a Telegram bot, we search for and launch BotFather in Telegram. We create a new bot using the /newbot command. We specify the name and username of the bot, the username must be unique, not repeat the existing ones in the database and end with the word "bot". After creating a Telegram bot, we can configure and edit it

if necessary using the menu. To connect the bot, its API Token is used, which is unique for each Telegram bot.

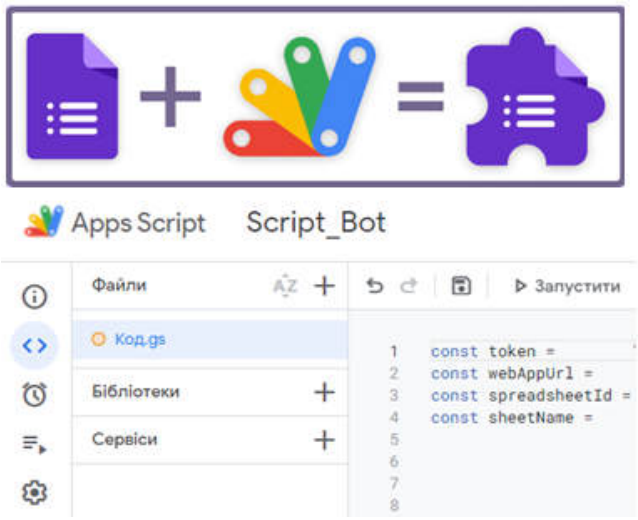

Figure 1. GAS scripting platform

Stage 2. Development of bot logic in Google Apps Script.

When developing the Telegram bot, we will use Google Apps Script (GAS), which is a scripting platform developed by Google for the development of light web applications on the Google Workspace platform (Fig. 1). GAS was originally developed by Mike Harm as a side project while working on Google Sheets. The framework is based on JavaScript 1.6, but also includes some parts from 1.7 and 1.8, as well as a subset of the ECMAScript 5 API. GAS projects run on Google's server-side infrastructure. According to GAS, it "provides simple ways to automate tasks at the intersection of Google products and third-party services." GAS is also a tool for writing extensions for Google Docs, Sheets, and Slides.

On the GAS platform, we create a new project where we set the variables token (token of the Telegram bot, which is used to communicate with the Telegram API), webAppUrl (URL of the web application, which is used as a webhook (Fig. 2) to receive incoming messages from Telegram , a webhook is a method of increasing or extending the functionality of a web page or web application with the help of user callbacks (callbacks)), spreadsheetId (the Google ID of the table in which the data will be entered), sheetName (the name of the sheet in the Google sheet in which the chat messages will be entered in Telegram), sheetName2 (the name of the sheet in Google Sheets in which user data will be entered).

When developing a Telegram bot, we use the following functions: setWebhook() (sets a webhook for a Telegram bot using the Telegram API), sendText(chat_id, text, keyBoard, firstName, lastName, currentDate) (sends a text message to the user using Telegram's sendMessage method API The function parameters include: chat_id (chat ID), text (message text), keyBoard (display keyboard), firstName (username), lastName (user's last name), and currentDate (current date/time)), doPost (e)

(the function processes incoming HTTP requests sent by the web application, receives data about the incoming message from Telegram, parses it and enters the necessary data into a Google table).

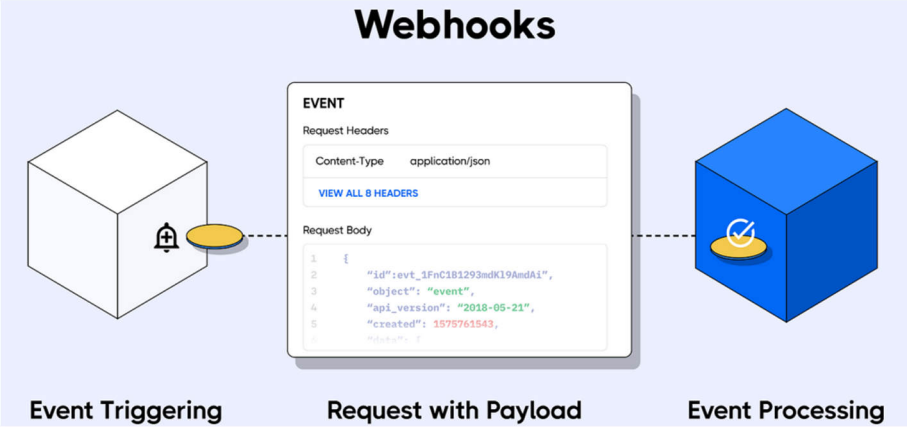

Figure 2. The webhook method

Also, we use the following objects: KEYBOARD_1, KEYBOARD_2 (objects represent keyboards (Fig. 3) for display in the Telegram chat, contain lines and buttons that can be pressed to interact with the bot).

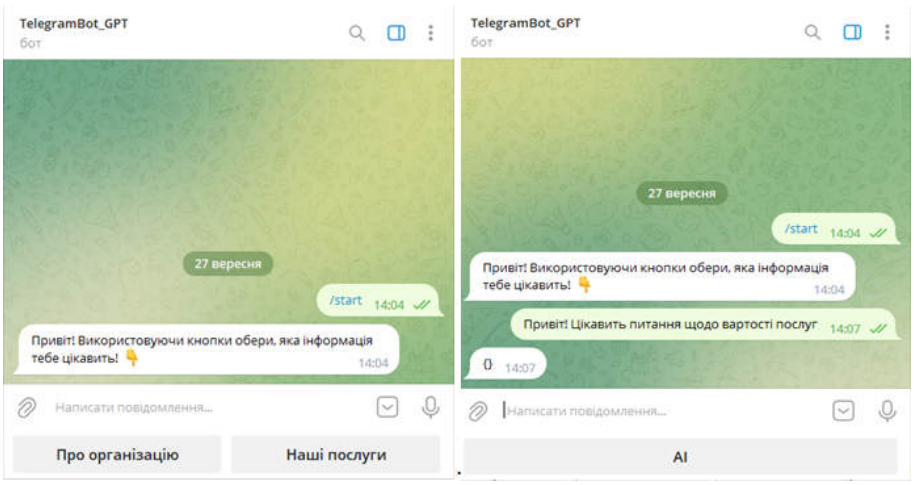

Figure 3. Telegram bot keyboards

Stage 3. Integration of AI.

To help users, we will use GPT-3 (Generative Pretrained Transformer 3) as an example, which is an autoregressive language model that uses deep learning to produce human-like text (Figure 4). It is the third-generation language predictive model in the GPT-n series, created by OpenAI, an AI research laboratory in San

Francisco [6]. Before the release of GPT-3, the largest language model was Microsoft's Turing NLG, introduced in February 2020, with a capacity of 17 billion parameters, or less than 10% of that of GPT-3 [7]. The quality of the text generated by GPT-3 is so high that it is difficult to distinguish it from text written by a human, which carries both advantages and risks [7]. Thirty-one OpenAI researchers and engineers presented the original paper on May 28, 2020, which introduced GPT-3. In their paper, they warned of the dangers of GPT-3's potential, and called for research to reduce the risk. David Chalmers, an Australian philosopher, described GPT-3 as "one of the most interesting and important AI systems ever made" [8].
To use GPT to help users with Telegram bot and connect GAS to it, there are several steps to follow:

Step 1. GPT integration (we access the service provided by GPT (for example, OpenAI GPT-3) and get an API key; we use the API key in GAS to interact with GPT, sending requests and receiving responses).

Step 2. Processing of user messages and replies (when a user sends a message to a bot in Telegram, GAS receives this message via webhooks, processes the user's message and extracts the necessary information).

Step 3. Sending requests to GPT and processing responses (we make requests to GPT, including the text of the user's message or other necessary information; we receive a response from GPT and process it for further use).

Step 4. Sending a response to the user (we create logic that sends a processed response to the user in Telegram via the Bot API).

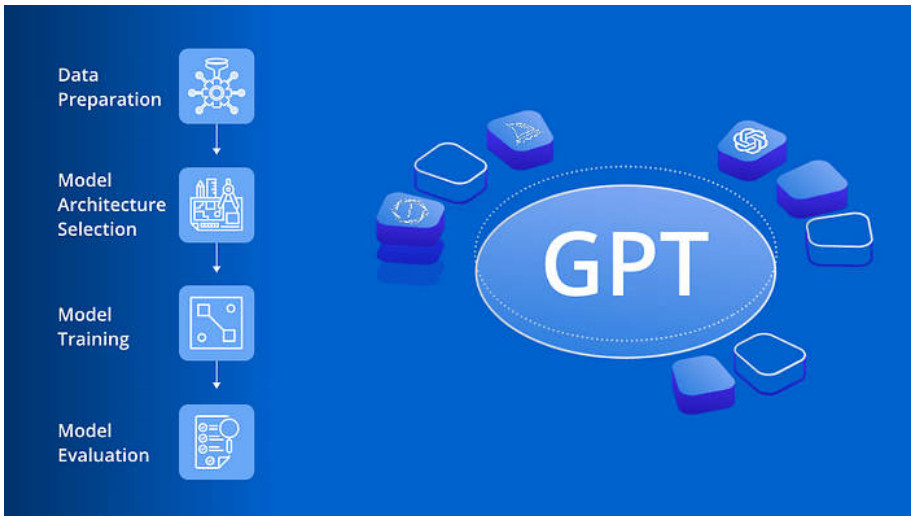

Figure 4. The GPT learning model

Stage 4. Formation of the database.

To form the database, a record will be made in Google Sheets (writeToGoogleSheet) using the writeToGoogleSheet function, which is responsible for recording user data and requesting it to Google Sheets (Fig. 5). The function adds a new line with the necessary data, in case of errors, it logs them. This allows the bot to interact with the user through Telegram, if necessary, call AI to generate responses and record information about users and their requests to the database.

	A	B	C	D
1	chat_id	firstName	lastName	currentDate
2	596535499	Viktor	Gnatyuk	19-9-2023 15:42:03
3	336387190	Leo		19-9-2023 17:32:41
4				
5				

Users

Figure 5. Google Sheet "Users"

Thus, a solution was developed to optimize the work of MSS O using VA (Telegram bot) and integration with GAS, Google Tables and GPT (GPT-3.5). Common features of this solution include the following: use of a Telegram bot (the VA is implemented as a Telegram bot that provides convenient communication with users through the Telegram platform), interaction with users (the bot interacts with users using buttons and text requests, allowing them to select information that they are interested in), generation of answers using GPT-3.5 (GPT-3.5 is used to expand the possibilities of VA, which allows generating more complex and informative answers), saving information in Google Tables (information about users and their requests is stored and managed in Google Tables, simplifying work with data and its analysis), use of GAS (use of GAS for bot logic programming and integration with Google Tables, which allows automating data processing and storage), optimization of MSS work (VA and integration with GPT-3.5 are aimed at optimizing user service , providing a quick and informative response to their inquiries). The source code of the developed software is available at the link [9].

These are the general features of the developed solution, which combines effective communication with users through Telegram, expanding the possibilities of generating answers with the help of AI, and effective management and analysis of data through Google Tables and GAS.

3. Conclusions

The developed solution expands the capabilities of existing MSS through the use of AI and the development of AI, in particular GPT-3.5, to ensure more effective and informative communication with users. Thus, an effective tool for automating and improving service processes has been developed, providing: improved user interaction, use of AI to improve responses, efficient data storage and analysis, and the possibility of automation thanks to GAS. In the future, it is planned to expand the

language model, improve the user interface, add an automatic language recognition module to support multiple languages and additional query analysis capabilities, develop algorithms that learn from user responses to provide personalized responses and improve the interaction experience, research and optimize data processing algorithms for faster and more efficient operation of the system with a large flow of requests. These scientific developments can improve the efficiency, accuracy and user experience of MSS using VA.

REFERENCES

1. KLEINROCK L.: Queueing Systems, Volume I - Theory. Wiley 1976, 417 p.
2. GELENBE, E., MITRANI, I.: Analysis and Synthesis of Computer Systems. Academic Press. London; New York (1980): Academic Press. 239 p.
3. ANANTHANARAYANAN G., et al.: CloudScale: Elastic Resource Allocation for Cloud Computing Environments. ACM (2010).
4. ZHANG H., HOU J. C.: Queue Length Estimation and Call Admission Control in Differentiated Services Networks. IEEE/ACM Transactions on Networking, 13(2005)2, 400-413.
5. LI W., LI Y.: Learning Automata-based QoS-aware Web Service Selection. IEEE Transactions on Services Computing, 2(2009)1, 48-61.
6. SHEAD S.: Why everyone is talking about the A.I. text generator released by an Elon Musk-backed lab. CNBC. *https://ramaonhealthcare.com/why-everyone-is-talking-about-the-a-i-text-generator-released-by-an-elon-musk-backed-lab/*
7. SAGAR RAM: OpenAI Releases GPT-3, The Largest Model So Far. Analytics India Magazine. *https://analyticsindiamag.com/open-ai-gpt-3-language-model/* June 3, 2020.
8. CHALMERS D.: GPT-3 and General Intelligence. У Weinberg, Justin. Daily Nous. Philosophers On GPT-3 (updated with replies by GPT-3). 30/07/2020.
9. Source code GPTbot. Internet resource Github: *https://github.com/ViktorGnatyuk/My_code/blob/main/GPTbot*