IRSTI 50.01.85

A. Amankossova[1], C. Turan[2]
[1,2]Suleyman Demirel University, Kaskelen, Kazakhstan

IMPLEMENTATION OF A REAL-TIME ALERT-NOTIFICATION SYSTEM FOR DATA MONITORING IN THE FINANCIAL INDUSTRY

Abstract. An alert notification system that operates in real-time is crucial for data monitoring, as it enables financial institutions to detect and address possible compliance issues promptly, preventing them from escalating into more significant problems. In the event that there are issues with the data, we may not realise that some parameter values are missing. In such instances, automatic alerts are unquestionably the most effective system. The use of an automated system to notify the responsible parties will benefit any organization's digitalization and reduce the burden on employees who must monitor manually. The article provides an examination and contrast of real-time alert automation techniques implemented through the programming languages of R and Python. The analysis encompasses the problems and limitations of the different methods. This study contributes to the field of automated notifications in the financial industry by evaluating different methods.

Keywords: automation, alert-notification system, financial industry, real-time, programming language

***

Аңдатпа. Нақты уақыт режимінде жұмыс істейтін ескерту хабарландыру жүйесі деректерді бақылау үшін өте маңызды, өйткені ол қаржы институттарына мүмкін болатын сәйкестік мәселелерін тез арада анықтауға мүмкіндік береді және олардың маңыздыр мәселелерге ұласуына жол бермейді. Деректерге қатысты мәселелер туындаған жағдайда, біз кейбір параметр мәндерінің жоқ екенін түсіне алмаймыз. Мұндай жағдайларда автоматты ескертулер ең тиімді жүйе екені сөзсіз. Жауапты тұлғаларды хабардар ету үшін автоматтандырылған жүйені пайдалану кез келген ұйымның цифрландыруына пайдалы болады және қолмен бақылауға тиіс қызметкерлерге жүктемені азайтады. Мақалада R және Python бағдарламалау тілдері арқылы жүзеге асырылатын нақты уақыттағы ескертулерді автоматтандыру әдістерінің сараптамасы қарастырылған. Талдау әртүрлі әдістердің проблемалары мен шектеулерін қамтиды. Жалпы алғанда, бұл зерттеу әртүрлі әдістерді салыстыру арқылы ескерту хабарландыру саласына ықпал етеді.

Түйін сөздер: автоматтандыру, ескерту-хабарландыру жүйесі, қаржы индустриясы, нақты уақыт режимі, бағдарламалау тілі.

***

Аннотация. Система оповещения, работающая в режиме реального времени, имеет решающее значение для мониторинга данных, поскольку она позволяет финансовым учреждениям быстро обнаруживать и устранять возможные проблемы с соблюдением требований, предотвращая их перерастание в более серьезные проблемы. В случае возникновения проблем с данными мы не можем понять, что некоторые значения параметров отсутствуют. В таких случаях автоматические оповещения, несомненно, являются наиболее эффективной системой. Использование автоматизированной системы для уведомления ответственных сторон принесет пользу цифровизации любой организации и снизит нагрузку на сотрудников, которые должны осуществлять мониторинг вручную. В статье рассматриваются и сравниваются методы автоматизации оповещений в реальном времени, реализованные с помощью языков программирования R и Python. Анализ охватывает проблемы и ограничения различных методов. В целом, это исследование вносит свой вклад в область автоматизированных уведомлений в финансовой отрасли, оценивая различные методы разработки.

Ключевые слова: автоматизация, система оповещения, финансовая индустрия, режим реального времени, язык программирования

I. Introduction

The efficiency of any organization's activities is determined by a variety of circumstances, including the correct monitoring of each key performance metric, the resolution of emergent issues, and the timely implementation of orders and requests from higher-level organizations. The financial industry is a critical sector that demands continuous monitoring of data to ensure conformity with regulations and detect anomalies or errors. This area is highly regulated, and compliance monitoring is crucial to ensuring that financial institutions comply with rules and laws. Data compliance monitoring is the process of verifying that financial institutions adhere to regulatory standards and promptly detect and address any issues that arise. One of the significant challenges in this monitoring is identifying data problems that may cause non-compliance. One of the main factors in approving a loan in the financial sector is the applicant's age, as it is regulated by law. As a result, this option should never be empty. Otherwise, the bank will face legal consequences. Accordingly, this research is set to send automatic notifications if the client's age is not supplied or if this parameter is empty. These data problems may include errors, anomalies, or precarious activities that could lead to severe financial losses, reputational damage, or regulatory sanctions. Between the occurrence of a significant event and the pursuit of activities, a significant amount of time elapses. With the growth of big data, the need for real-time alert and notification systems for data monitoring has become even more significant. Real-time notification systems

are essential components of data compliance monitoring, allowing financial institutions to quickly detect and respond to potential issues. These systems provide immediate alerts to compliance teams, enabling them to take corrective action and prevent non-compliant activities from escalating. Developing automated alerts for responsible individuals is one technique to enhance the tracking of an organization's business processes. An automated email notification process is considered one of the most effective strategies to send an operational "push notification" email informing responsible employees that there is currently a data problem.

This research work aims to evaluate and compare various methods for developing automated notifications in real-time. The primary goal of this study is to identify the most effective and efficient approach for creating automated notification systems that can provide real-time alerts for various events and situations in the financial industry. The research will investigate different techniques that are available for building real-time notification systems using the Python and R programming languages. The study will also analyse the advantages and limitations of each method. The findings of this study are expected to contribute to the development of more effective and efficient real-time notification systems, which can enhance the quality of service and decision-making.

There is a lack of work on automated alert systems via email in real-time. Many of them are not automated or not in real-time. This paper has eliminated this problem. Research papers by Robert P. "Implementing an email-based notification system for large-scale system resources" [1] and S. K. N. A. Rahim and N. R. P. Ismail "Automated Attendance Management and Alert System" [2] motivate this article to explore this field even further. According to Robert's study, real-time alert-notification systems are critical components of monitoring large-scale system resources, as they enable organisations to detect and respond to potential violations promptly. The study evaluated the performance of a real-time alert and notification system developed using the Python programming language and found that the system was effective as an email-based notification system. While both works involve the implementation of notification systems, they are focused on different applications.The work by Robert P. on implementing an email-based notification system is focused on large-scale system resource management, specifically the notification of system administrators when certain events or thresholds are reached, such as when system resources are low or when hardware failures occur. This notification system is designed to improve system efficiency and reduce downtime.

In another study, Rahim and Ismail developed an automated alert system for monitoring the attendance of students. The system is designed to automate the attendance tracking process, reduce errors, and improve efficiency in managing attendance data. The system also features an alert system that sends notifications to users when certain attendance thresholds are breached. The study

showed the effect of the alert on the rapid resolution of errors. On the other hand, this research work is focused on the development of an alert system for financial data monitoring. The system is designed to monitor financial data in real-time and send notifications to users when certain events or conditions occur. This system is designed to assist analysts in making decisions by providing timely and accurate information.

Another work "Sending Email with Python" by Joska de Langen [3] is a comprehensive guide to sending emails using the Python programming language. It covers the basics of how emails work, how to authenticate with email servers, and how to construct and send emails with attachments. This article is relevant to the topic of automated email notifications as it provides practical guidance on how to use Python to send emails programmatically, which is a common approach for sending automated notifications.

The literature suggests that automated notification systems are effective tools for monitoring in different industries and domains. The studies have shown that different programming languages, such as R and Python, can be used to develop real-time alert and notification systems that are effective in automating and reducing response time. However, the literature also indicates that the effectiveness of alert notifications depends on various factors, such as the quality of the data, the type of technique used, and the nature of the business process being monitored.

Further research is needed to investigate the effectiveness of real-time alert and notification systems in different scenarios and to identify the most effective approaches for developing such systems. The paper will describe the process of implementing, testing, and evaluating a real-time alert and notification system for compliance monitoring in the financial industry using R and Python. The implementation of a system that allows notification duties to be completed automatically has a high likelihood of lowering the risk of problems with a business metric, in this case, problems with missing value identification. This will provide a detailed description of the system architecture and the mechanisms for generating and sending alerts to relevant stakeholders.

Overall, this study contributes to the field of automated notifications in the financial industry by providing a thorough evaluation of different methods in the Python and R programming languages. Specifically, this study fills a gap in the existing literature by comparing and contrasting the efficacy of various approaches to real-time alert-notification systems as well as identifying the strengths and limitations of each method. By conducting a comprehensive analysis of the various tools available, this study provides valuable insights into the best practises for developing an effective automated notification system that can help financial institutions detect and respond to potential compliance issues in a timely and efficient manner. Furthermore, the findings of this study have practical implications for financial institutions seeking to improve their data monitoring and compliance efforts, as they can use the results of this research to

inform their decision-making and select the most appropriate approach for their specific needs.

II. Methodology

The programming languages R and Python were chosen as the development environment for creating and designing the implementation of the alert system, or, in other words, notifications.

The R programming language emphasises its flexibility, versatility, and extensibility. Its wide range of features and capabilities make it an ideal tool for a wide range of tasks with data, from simple exploratory analysis to complex machine learning projects [4]. Moreover, its open-source nature and active community of developers and users make it a vibrant and constantly evolving language. Python is a versatile programming language that can be used for a wide range of scientific and data analysis tasks [5]. It also provides a range of tools and libraries that can be used to send email notifications.

The data consisted of several variables, including demographic information such as age, gender, and geography. These variables could be quantitative, such as numerical measurements of a particular phenomenon, or qualitative, such as categorical variables that describe characteristics or opinions. The open database provided some raw data that took the following shape for this article (see Figure 1):

Id	Time_stamp	CreditScore	Age	Geography	Gender	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
1	26.01.2023 12:36	619	42	France	Female	2	0	1	1	1	101348.88	1
2	26.01.2023 12:36	608	41	Spain	Female	1	83807.86	1	0	1	112542.58	0
3	26.01.2023 12:36	502	42	France	Female	8	159660.8	3	1	0	113931.57	1
4	26.01.2023 12:36	699	39	France	Female	1	0	2	0	0	93826.63	0
5	26.01.2023 12:36	850	43	Spain	Female	2	125510.82	1	1	1	79084.1	0
6	26.01.2023 12:36	645	44	Spain	Male	8	113755.78	2	1	0	149756.71	1
7	26.01.2023 12:36	822	50	France	Male	7	0	2	1	1	10062.8	0
8	26.01.2023 12:36	376	29	Germany	Female	4	115046.74	4	1	0	119346.88	1
9	26.01.2023 12:36	501	44	France	Male	4	142051.07	2	0	1	74940.5	0
10	26.01.2023 12:36	684	27	France	Male	2	134603.88	1	1	1	71725.73	0
11	26.01.2023 12:36	528	31	France	Male	6	102016.72	2	0	0	80181.12	0
12	26.01.2023 12:36	497	24	Spain	Male	3	0	2	1	0	76390.01	0
13	26.01.2023 12:36	476	34	France	Female	10	0	2	1	0	26260.98	0
14	26.01.2023 12:36	549	25	France	Female	5	0	2	0	0	190857.79	0
15	26.01.2023 12:36	635	35	Spain	Female	7	0	2	1	1	65951.65	0
16	26.01.2023 12:36	616	45	Germany	Male	3	143129.41	2	0	1	64327.26	0
17	26.01.2023 12:36	653	58	Germany	Male	1	132602.88	1	1	0	5097.67	1
18	26.01.2023 12:36	549	24	Spain	Female	9	0	2	1	1	14406.41	0
19	26.01.2023 12:36	587	45	Spain	Male	6	0	1	0	0	158684.81	0

Figure 1: A sample of the data

Figure 2 depicts the three-level architecture that was adopted to ensure the system's functionality. There is no direct link between the thin client layer and the database. The application logic is implemented in a programming language, and the information is stored in a database. The database was created and edited using the relational DBMS Oracle PL/SQL. The primary tables used to store details about events, employee contacts, and personal information are found in the alert system. The time of the most recent notice is then executed, and if the condition returns true, the letter is created and delivered to the employee at the supplied contact.

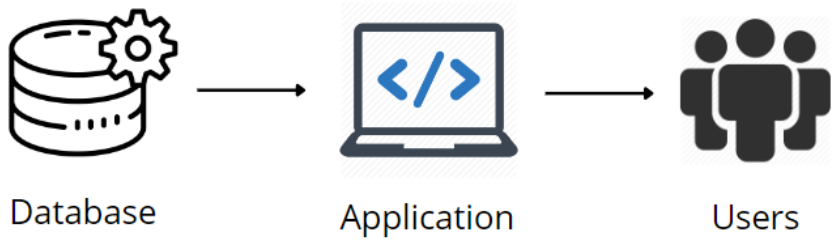

Figure2: Three-level architecture of the alert system

Here are some of the techniques that can be used to send automated email notifications.

1. SMTP

The Simple Mail Transfer Protocol (SMTP) is a protocol used for sending emails over the internet [6]. Python and R provide a built-in SMTP library that can be used to send emails programmatically. The smtplib library in Python can be used to create an SMTP connection, authenticate with an email server, and send emails. This technique can be used to send email notifications from a scientific application or workflow. However, it requires knowledge of the email service provider's SMTP settings and programming skills in R. It is important to follow best practises for secure and reliable email notification sending and to ensure that the SMTP configuration is properly tested and maintained.

2. Cron Jobs

Cron is a time-based job scheduler in Unix-like operating systems that can be used to schedule and automate repetitive tasks [7]. Python and R scripts can be scheduled as cron jobs to send email notifications at specified intervals. This technique can be used to automate email notifications from applications and workflows. It requires knowledge of Unix-based operating systems and cron configuration, as well as programming skills.

3. APIs

Using APIs (Application Programming Interfaces) is a powerful way to send notifications programmatically and in real-time [8]. APIs allow developers to integrate their applications with third-party services and leverage their functionalities, including sending notifications via email. Many email service providers offer APIs using services like Mailgun, SendGrid and Amazon SES that can be used to send email notifications programmatically. Python libraries such as Requests, urllib and R packages such as httr, sendgridr, and aws.ses can be used to make API requests and send email notifications. This technique can be used to send email notifications from scientific applications that are hosted on cloud platforms.

III. Results

In this study, real-time alert-notification system implementation models satisfying three requirements were taken into consideration.

1. Alert-notification

2. Automated process
3. Real-time observation

First, different methods of real-time alert-notification systems using Python and R programming languages were implemented and tested. The implementation process involved designing and developing notification systems and testing them in a simulated financial environment.

Here's a comparison table between Python and R for sending automated notifications (Table 1):

Table 1: Python and R comparison table

Criteria	Python	R
Speed of Execution	Python is generally faster than R when it comes to executing code.	R can be slower than Python, especially for larger datasets or more complex calculations.
Flexibility	Python is more flexible than R when it comes to handling different data formats.	R has a more specific focus on statistical computing and may not be as flexible for other tasks.
Ease of Use	Python has a simpler syntax and is easier to learn for beginners	R has a steeper learning curve, but it is more specialized for statistical computing.
Availability of APIs	Python has a vast number of APIs and libraries available, including Mailgun and Sendgrid.	R has a smaller number of APIs available, but it does have access to Amazon SES and other APIs.

Both Python and R are capable of sending automated notifications, each with their own strengths and weaknesses. The Python-based system had a shorter response time compared to the R-based system. Python has a simpler syntax and is more widely used, with more APIs and libraries available. R is more specialised for statistical computing and has a wider range of statistical libraries available.

Next, the effectiveness of techniques for developing real-time alert-notification systems for compliance monitoring in the financial industry was compared and evaluated. This comparison highlights the advantages and limitations of each method. The results of the evaluation revealed that while all systems were effective in alerting users to potential compliance issues in real-time, there were some differences in their performance.

The results of the evaluation of different methods of automated notifications using Python and R programming languages are presented in Table 2. The effectiveness and efficiency of alert-notification systems were evaluated by measuring key performance indicators, including monthly costs, reliability, speed of execution, advantages, and limitations.

Table 2: A comparison of sending notifications methods

Methods		Cost	Time	Reliability	Limitations	Advantages
SMTP		Free	<1 second	High	1.Authentication and security 2.Deliverability 3.Scalability	1. Immediate delivery; 2. Availablity 3.Cost
Using cron jobs		Free	< 1 second	Medium	1. Limited functionality; 2. Dependence on the operating system; 3. Debugging;	1. Cost
Using APIs	Amazon SES	$24.95 per month	< 1 second	High	1.Learning curve; 2.Limited support; 3. Cost;	1. Scalability 2. Deliverability
	Mailgun	$35 per month			1.Learning curve; 2.Limited support; 3. Cost; 4. Deliverability issues	1.Easy integration 2.Detailed analytics 3.Advanced features
	Send Grid	$14.95 per month			1.Learning curve; 2.Limited support; 3. Cost; 4. Deliverability issues	1. Scalability 2.Advanced features

These techniques have their own advantages that make them suitable for different use cases. But each method has its limitations. SMTP is susceptible to

spam and phishing attacks due to its lack of robust authentication and security features. It can be subject to delivery failures if the recipient's email address is invalid, the email is flagged as spam. It may not be suitable for large-scale email campaigns due to its limitations. But the system uses short message formats. Therefore, it is not a restriction to us. Cron jobs are limited to running pre-defined scripts at scheduled times, making them less flexible than other notification methods. It relies on the underlying operating system, which may introduce compatibility issues or require additional maintenance. Debugging cron jobs can be challenging, particularly when trying to diagnose issues with scripts that run at scheduled intervals. Using APIs for sending notifications provides a high degree of flexibility and control over the email notifications. It allows developers to customise the content and formatting of the email notifications and trigger them programmatically in real-time. However, it requires programming skills and knowledge of APIs, and some email service providers charge usage fees for their APIs. It is important to consider the cost-effectiveness of using an email service provider that offers an API, as well as the time and resources required to develop and maintain the API integration. They can be more expensive than other notification methods because the cost varies depending on the number of emails sent and the level of service used. For example, the minimum number of own addresses in the Amazon SES API, which is a minimum amount of $24.95 x 256 = $6387.20 per month, $6387.20 * 12 months = $76646.4 per year. SMTP can help save money for organisations that need to send a large volume of emails regularly. By using an SMTP server, businesses can avoid paying for expensive email marketing services or other third-party software. Additionally, despite the limitations, SMTP allows financial organisations to have greater control over their email data, which can be particularly important for regulatory compliance purposes. This level of control and compliance can be particularly important for organisations in industries with strict regulatory requirements, such as finance.

Based on the evaluation of the different methods for automated notification, it can be concluded that SMTP offers significant cost-saving advantages compared to other methods such as Mailgun, SendGrid, and Amazon SES API. While SMTP may have limitations in terms of security and scalability, the research shows that it is not significantly worse than other methods in terms of reliability and time of execution. Therefore, SMTP can be a viable option for organisations looking to implement automated notification systems while keeping costs low.

IV. Conclusions

In conclusion, the research work will contribute to the growing body of literature on the use of real-time alert-notification systems for data compliance monitoring and provide insights into the best practices for implementing such systems using R and Python. The evaluation of the real-time alert-notification system for data

monitoring in the financial industry has shown that SMTP is a more suitable method. This research has highlighted the importance of implementing an effective alert-notification system for financial institutions to comply with regulations and ensure the security of their data. This can help mitigate the potential negative impacts of data breaches and enhance overall security measures within the financial industry.

In future work, I want to improve this service and send alerts not only when there is a data problem, but also in situations of anomalous business metrics.

References:

1 Robert Poenaru, “Implementation of an email-based alert system for large-scale system resources” // RoEduNet Conference: Networking in Education and Research (20), 2021

2 S. K. N. A. Rahim, N. R. P. Ismail, F. A. Razak, I. Z. Zulkifli, N. H. Jamian , N. F. Razi and N. H. Mohammad, “Automated attendance management and alert system” // Journal of Fundamental and Applied Sciences, 2017

3 Joska de Langen “Sending Email with Python.” Real Python, 21 May 2021, realpython.com/python-send-email/.

4 Daniel W., Bhargav A., John K., Lemuel R., “Integrating R efficiently to allow secure, interactive analysis within a clinical data warehouse” // International useR Conference (8), 2012

5 Lee, Youngsook, and Jinhwa Cho. "The Influence of Python Programming Education for Raising Computational Thinking." International Journal of U- and E-Service, Science and Technology, vol. 10, no. 8, 2017, pp. 59-72.

6 P. Hoffman, “Smtp service extension for secure smtp over transport layer security,” RFC 3207, Feb. 2002.

7 Le, Nhi. "Automated Jira backlog prosess using Python." (2022).

8 Ofoeda, Joshua, Richard Boateng, and John Effah. "Application programming interface (API) research: A review of the past to inform the future." International Journal of Enterprise Information Systems (IJEIS) 15.3 (2019): 76-95.