Woody International Publish Limited
An Multidisciplinary Academic Journal Publisher
Journal of Artificial Intelligence and Information, Volume 2, 2025
https://www.woodyinternational.com/

Implementation of An Automated Intelligence Collection Framework Based on Go Language

Youming Zhang[1], Ke Chen[2]

School of Computer and Software, Chengdu Jincheng University, Chengdu, Sichuan 611731, China

Abstract: ***At present, search engine hacking is still a highly threatening attack method to security. Through search engines, victims' website backend or source code, and even users' personal privacy information can be directly obtained from the network. In order to minimize this security threat as much as possible and improve the efficiency of maintaining network security, an automated intelligence collection framework based on Go language is proposed, mainly through Bing and Google search engines for intelligence collection. This framework utilizes the concurrency feature of Go language, greatly improving the performance of crawlers and accelerating the speed of intelligence collection.***

Keywords: Crawling technology; Go language; Intelligence gathering; Network security; Search Engines.

1. INTRODUCTION

Search engines are always a powerful double-edged sword. Hacker attacks are one of the main threats. Without restrictions, many private information exposed on the internet can be accurately searched through special syntax structures. Most of the time, hackers can always break through the defense line through these inconspicuous information, attack the inside and cause huge damage. But for those who maintain security, searching for and maintaining these small pieces of information one by one is a huge workload, so understanding and learning from the methods of bad people is the key to solving problems more efficiently. The automated intelligence gathering framework based on Go language can improve the work efficiency of security personnel and reduce the workload of intelligence gathering to a certain extent. Several papers demonstrate the effectiveness of deep learning in image processing and computer vision tasks. Yan et al. (2024) focus on improving image super-resolution using convolutional neural networks. Xu et al. (2024) leverage YOLOv5 for real-time crown-of-thorns starfish detection in automated surveillance, highlighting the practical applications of object detection. Tian et al. (2024) propose an improved U-Net model for accurate brain tumor segmentation, demonstrating the use of deep learning in medical image analysis. Chen et al. (2022) present a one-stage object referring method that incorporates gaze estimation, pushing the boundaries of object recognition. Chen et al. (2020) utilize deep learning for automated defect grading in printed materials, illustrating its use in industrial quality control. These studies showcase the versatility of deep learning across various image-related tasks. The application of big data and AI in finance is a recurring theme. Ravi and Kamaruddin (2017) provide an overview of the opportunities and challenges associated with big data analytics in smart financial services. Eltweri et al. (2021) specifically address the use of big data for fraud detection and risk management within the real estate industry. Murugan (2023) focuses on large-scale data-driven financial risk management using machine learning strategies. VenkateswaraRao et al. (2023) present a big data analytics-based credit investigation and risk management system for commercial banking. Bi et al. (2024) develop a financial intelligent risk control platform using big data analysis and deep machine learning. Tekaya et al. (2020) and Hasan et al. (2020) offer broader overviews of big data's impact on the financial sector. Shakya and Smys (2021) highlight the use of big data analytics for improved risk management and customer segmentation in banking. These studies collectively illustrate the significant role of data-driven techniques in improving efficiency, security, and decision-making within the financial industry. While not the primary focus of many papers, NLP plays a significant role in some studies. Ren (2024) presents a novel approach for role-oriented dialogue summarization, improving the quality of summaries generated from conversations with multiple participants. Lu (2024) employs machine learning to enhance chatbot user satisfaction, demonstrating the impact of AI on user experience. Wu (2024) utilizes large language models for semantic parsing in an intelligent database query engine. These studies highlight the increasing importance of NLP in improving human-computer interaction and information access. The foundational work by Jurafsky and Martin (2007), Bethard et al. (2008), and Nadkarni et al. (2011) provides crucial background information in the field of NLP. Ren (2024) further improves Seq2Seq models for dialogue summarization. Lin et al. (2024) offer a comprehensive review of precision anesthesia for high-risk surgical patients, highlighting the potential of advanced technologies in improving patient outcomes. Qi and Liu (2024) focus on designing a sales forecasting system using Hadoop-based big data analysis. Wang et al. (2024) utilize graph neural networks for building a recommendation system for football formations. Zheng et al.

(2024) investigate improving the efficiency of deep learning optimizers. Li et al. (2024) examine the impact of policies promoting the integration of technology and finance on green innovation. Chen et al. (2024) discuss AI-driven threat detection in cybersecurity. Zhu et al. (2024) work on adversarial methods for sequential recommendations. Xie et al. (2024) focus on legal citation text classification using Conv1D. Li (2024) applies multimodal data and multi-recall strategies to enhance e-commerce product recommendations. Xu et al. (2024) study experience management tools in the electric vehicle market. Chen et al. (2024) explore computerized data mining techniques. Shen et al. (2024) present a dynamic resource allocation strategy for cloud-native applications. Chen and Bian (2019) present a streaming media live broadcast system. Liang and Chen (2019) introduce a high-performance dynamic service orchestration algorithm.

2. BRIEF INTRODUCTION

2.1 Functional Framework

The automated intelligence gathering framework based on Go language proposed in this article includes the main functions of search engine syntax gathering intelligence, web JS script crawling and analysis, web screen screenshot, and custom search engine syntax. The specific functional structure diagram is shown in Figure 1.

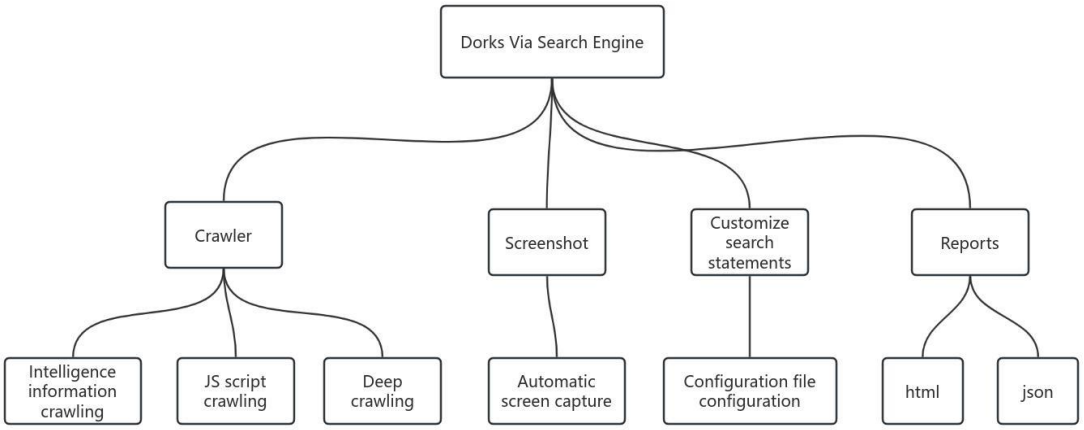

Figure 1: Framework Function Diagram

2.2 Framework Function Introduction

Command line interface: Function selection and parameter initialization can be performed through the command line. Compared to a beautiful interface, the command line reduces unnecessary interface rendering and can be run directly in the shell, making it more convenient and efficient.

Custom search statement rules: By specifying the file path through the command line, the framework automatically initializes the program with user-defined rule statements. Considering the uncontrollability of user input, a fixed file content template was adopted to facilitate program parsing of file content. At the same time, the framework will have default search engine syntax, which is convenient for ordinary users to use directly at any time without considering the issue of search engine syntax. During the initialization of the framework, the latest search syntax rules will be pulled from the specified address to complete the syntax information configuration locally.

Crawler obtains intelligence information: Obtains the target domain or site specified by the user, calls the search engine crawler interface of the framework for information retrieval, and the search engine will perform the first layer of information filtering and processing on the retrieved information. The framework will further process and analyze the retrieved information based on the retrieved information, and finally provide the user with a concise information report.

JS script crawling: By default, this framework only crawls information retrieved by search engines. If the user specifies deep crawling, the framework will further explore the retrieved site. If the target is a site, JS often exposes

a lot of private information about the site. Therefore, the program will crawl JS scripts based on configuration information to facilitate analysis by other JS script analysis programs.

Web page screenshot: Usually, the effect of searching for information by opening a web page alone is not high, so when crawling the information retrieved by search engines, a screenshot of the web page will be generated at the same time, which is convenient for directly viewing the image form of the retrieved information and can also improve the efficiency of finding and discovering attack surfaces. Headless Chrome technology combined with the Chrome framework is used for web page screenshot operations.

Report generation: Automatically generate JSON or HTML file format reports based on the information retrieved by the framework and the command parameters selected by the user when executing the program. It also provides the function of displaying the results directly on the screen without generating a report file.

2.3 Introduction to Core Technologies

2.3.1 Search engine syntax

The implementation of information retrieval functions mentioned in this article is based on search engine syntax, mainly using the search engine's boolean logical retrieval syntax, special retrieval syntax, and advanced search syntax, such as the special retrieval syntax's import and alliance, site, inurl, intent, advanced search syntax's filetype, info, and other commonly used retrieval syntax [4]. By combining the above search engine syntax in different ways, the function of information filtering can be achieved. For example, using site syntax to specify the target range, which can be a domain name or a precise site, and then combining advanced syntax initle to query the webpage title information of the retrieved information, inurl to query the link category of the webpage, filetype to specify the search file type filtering, such as. pdf,. sql,. xls,. doc and other files that may contain important information. After layer by layer filtering and filtering by search engine syntax, accurate target intelligence information can be obtained. This search engine syntax framework is also used in the default search engine rules of this automated intelligence gathering framework, which can accurately obtain relevant leaked information of the target.

2.3.2 Web Screen Capture Function

The webpage screenshot function mentioned in this article adopts the popular headless chrome browser in development, which has a faster loading speed compared to interface browsers. The chromedp framework was used in the actual implementation process of the framework. This framework is a faster and simpler Golang library for calling browsers that support the Chrome VNet protocol. Set tasks through its Tasks structure and use the CaptureScreenshot() function on the page interface to take screenshots. The combination of the two not only greatly reduces the difficulty of feature development, but also improves the efficiency of screenshot functionality.

2.3.3 Crawling Technology

The core modules used in the crawling part of this framework are the native web/http library and goquery library of the Go language, as well as other related crawling technology modules. The net/http module is mainly used to simulate the browser sending requests and receiving response data, while the goquery library is used to parse the HTML data returned by the net/http library response. Through the coordination of the two libraries, the basic crawler result processing is completed. The program then calls the two libraries for deep JS script crawling based on user input commands, and finally saves the results locally for easy subsequent calling and processing. In the crawler technology module, the concurrency feature of Go language is still used, allowing all crawling threads to execute concurrently and maximizing CPU utilization.

2.3.4 Automatic report generation technology

The final report generated by this framework is in two forms: JSON and HTML. The JSON format uses the encoding/json library native to the Go language, while the HTML format uses file reading and writing in conjunction with the GoHTML library. The GoHTML library can automatically generate HTML code through programming in the Go language. The JSON format facilitates quick parsing or calling by other programs, while HTML allows users to intuitively view intelligence collection results and perform data analysis.

3. DESIGN CONCEPT AND IMPLEMENTATION

3.1 Functional Process

This framework exists in the form of a command-line interface, where users input parameters and custom configuration files through the command line. The framework automatically parses the configuration files and commands entered by the user, and then collects intelligence through search engines based on the commands. Regardless of the final collection situation, a final report will be generated. The overall functional operation process is shown in Figure 2.

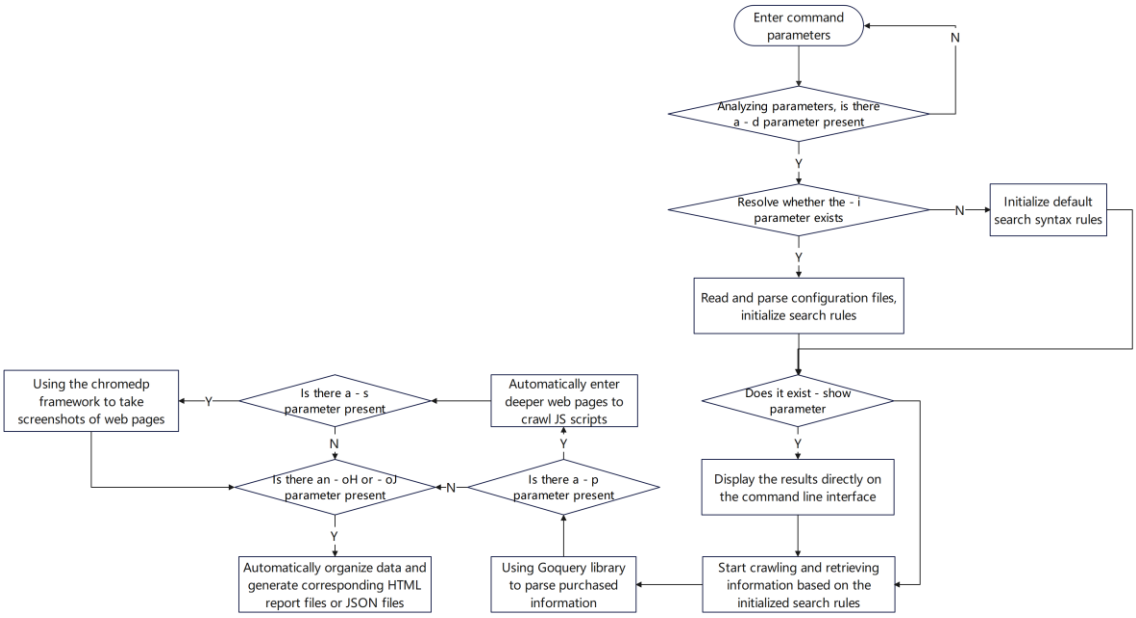

Figure 2: Framework Function Flow Chart

3.2 Implementation steps

3.2.1 Command line interface and parsing parameters

The command line of this framework is implemented using the CLI command line framework, and users complete operations through different parameters. In this framework, the - d parameter, which refers to the target domain name or URL, is a mandatory option. The search engine syntax rules allow users to use the default syntax or specify custom syntax rules through the - i parameter. The configuration file can be parsed using the YAML parsing library in the Go language. The combination of their values will provide a basic reference for subsequent operations. The main steps for implementing command-line functions are:

Step 1: Declare the Flag and Command command line operation modes by initializing the APP structure in the CLI structure in the CLI command line library. Flag is the global operation instruction, and Command is the command operation. In the APP structure, the Command property accepts a [] * cli Command, Flag accepts [] cli Flag enables different command-line operations.

Step 2: Map the value of Flag to the Options structure, which is a custom structure used to store the command parameter values entered by the user.

Step 3: Encapsulate this initialization process into the ParseOptions() function and use * Options as the return parameter for easy calling by the main package of the framework.

3.2.2 Intelligence Information Retrieval and Data Analysis

The intelligence retrieval process of this framework will retrieve and process intelligence information based on parameters parsed from the command line, custom syntax rules provided by the user, or default rules of the program. The information retrieval steps of the framework are as follows:

Step 1: Use the net/http library to simulate sending requests. The header mainly includes search syntax rules and the URL, User Agent, header, etc. concatenated with the user specified domain name or site.

Step 2: Accept the response data of the request. Through the goquery library

The NewDocumentFromRead() function parses the HTML data returned by the response and uses its Find() function to specify the retrieval syntax rules, filter out different information, and classify it.

Step 3: If the user sets - p, which means deep crawling information, the program will automatically enter the URL parsed in step 2 and reuse the goquery library. By combining the. js suffix, http://, https://, and<script>tags to retrieve rules, and filtering out the URL address of the JS script through the Find () function of the goquery library.

Step 4: If the user also specifies the - s parameter, which is a webpage screenshot, configure the task through the Tasks structure of the chromedp framework; The Navigate() function specifies the target URL, with parameter values derived from the URL parsed by the initial crawler; The LayoutMetrics() function obtains the height and width of the target webpage; The CaptureScreenshot() function retrieves a screenshot of the retrieved target webpage.

3.2.3 Intelligence report generation

The report of this framework is mainly aimed at facilitating viewing and other program calls for parsing, so HTML and JSON formats are adopted. The program will ultimately generate different types of reports based on the user's selection. The report mainly focuses on intelligence information, including crawled URLs, JS script URLs, and webpage screenshots. The generation of HTML reports is dynamically generated by the gohtml library, which combines the Tag() function of the gohtml library with filtered data to dynamically generate HTML code. The generated HTML code is then written into the file through file reading and writing, ultimately generating a complete HTML report. The JSON report mainly utilizes the encoding/json library, which defines the structure of JSON nodes through the structure of go language. The Marshal() function in the JSON library is used for encoding, dynamically saving the results in JSON format and automatically generating JSON reports.

4. CONCLUSION

The search engine HACK is still a serious security issue in the current era. With the efficient retrieval of search engines, hackers can obtain accurate intelligence information in a very short time. The automated intelligence collection framework based on the Go language proposed in this article imitates the attack method of the search engine HACK, which can automate intelligence retrieval and help improve security. Users can use this framework for self retrieval and discovery, and take immediate security measures to remedy the situation. Of course, this framework still has certain shortcomings, such as content recognition and type differentiation of intelligence information. In future optimizations, emphasis will be placed on addressing information processing, enhancing the efficiency and accuracy of identifying intelligence information, and improving the information crawling efficiency and collection accuracy of the framework.

REFERENCES

[1] Ying Dong Network hacker attacks and prevention governance in the era of big data [J] Network Security Technology and Applications, 2021 (05): 68-70
[2] Yan, H., Wang, Z., Xu, Z., Wang, Z., Wu, Z., & Lyu, R. (2024, July). Research on image super-resolution reconstruction mechanism based on convolutional neural network. In Proceedings of the 2024 4th International Conference on Artificial Intelligence, Automation and High Performance Computing (pp. 142-146).
[3] Xu, G., Xie, Y., Luo, Y., Yin, Y., Li, Z., & Wei, Z. (2024). Advancing Automated Surveillance: Real-Time Detection of Crown-of-Thorns Starfish via YOLOv5 Deep Learning. Journal of Theory and Practice of Engineering Science, 4(06), 1–10. https://doi.org/10.53469/jtpes.2024.04(06).01

[4] Zheng Ren, "Balancing role contributions: a novel approach for role-oriented dialogue summarization," Proc. SPIE 13259, International Conference on Automation Control, Algorithm, and Intelligent Bionics (ACAIB 2024), 1325920 (4 September 2024); https://doi.org/10.1117/12.3039616
[5] Ravi, V., & Kamaruddin, S. (2017). Big data analytics enabled smart financial services: opportunities and challenges. In Big Data Analytics: 5th International Conference, BDA 2017, Hyderabad, India, December 12-15, 2017, Proceedings 5 (pp. 15-39). Springer International Publishing.
[6] Lin, S., Tan, H., Zhao, L., Zhu, B., & Ye, T. (2024). The Role of Precision Anesthesia in High-risk Surgical Patients: A Comprehensive Review and Future Direction. International Journal of Advance in Clinical Science Research, 3, 97-107.
[7] Eltweri, A., Faccia, A., & Khassawneh, O. S. A. M. A. (2021, December). Applications of big data within finance: fraud detection and risk management within the real estate industry. In Proceedings of the 2021 3rd International Conference on E-Business and E-commerce Engineering (pp. 67-73).
[8] Wang, Z., Zhu, Y., Li, Z., Wang, Z., Qin, H., & Liu, X. (2024). Graph neural network recommendation system for football formation. Applied Science and Biotechnology Journal for Advanced Research, 3(3), 33-39.
[9] Tian, Q., Wang, Z., Cui, X. Improved Unet brain tumor image segmentation based on GSConv module and ECA attention mechanism. arXiv preprint arXiv:2409.13626.
[10] Nadkarni, P. M. , Ohno-Machado, L. , & Chapman, W. W. . (2011). Natural language processing: an introduction. Journal of the American Medical Informatics Association Jamia, 18(5), 544.
[11] Bi, S., Lian, Y., & Wang, Z. (2024). Research and Design of a Financial Intelligent Risk Control Platform Based on Big Data Analysis and Deep Machine Learning. arXiv preprint arXiv:2409.10331.
[12] Bethard, S. , Jurafsky, D. , & Martin, J. H. . (2008). Instructor's Solution Manual for Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition (Second Edition).
[13] Li, S. (2024). Harnessing Multimodal Data and Mult-Recall Strategies for Enhanced Product Recommendation in E-Commerce.
[14] Xu Y, Shan X, Guo M, Gao W, Lin Y-S. Design and Application of Experience Management Tools from the Perspective of Customer Perceived Value: A Study on the Electric Vehicle Market. World Electric Vehicle Journal. 2024; 15(8):378. https://doi.org/10.3390/wevj15080378
[15] Qi, T., & Liu, H. (2024, September). Research on the Design of a Sales Forecasting System Based on Hadoop Big Data Analysis. In Proceedings of the 2024 2nd International Conference on Internet of Things and Cloud Computing Technology (pp. 193-198).
[16] Zheng, H., Wang, B., Xiao, M., Qin, H., Wu, Z., & Tan, L. (2024). Adaptive Friction in Deep Learning: Enhancing Optimizers with Sigmoid and Tanh Function. arXiv preprint arXiv:2408.11839.
[17] Li, L., Gan, Y., Bi, S., & Fu, H. (2024). Substantive or strategic? Unveiling the green innovation effects of pilot policy promoting the integration of technology and finance. International Review of Financial Analysis, 103781.
[18] Chen, H., Shen, Z., Wang, Y., & Xu, J. (2024). Threat Detection Driven by Artificial Intelligence: Enhancing Cybersecurity with Machine Learning Algorithms.
[19] Zhu, Z., Wang, Z., Wu, Z., Zhang, Y., & Bo, S. (2024). Adversarial for Sequential Recommendation Walking in the Multi-Latent Space. Applied Science and Biotechnology Journal for Advanced Research, 3(4), 1-9.
[20] Lu, J. (2024). Enhancing Chatbot User Satisfaction: A Machine Learning Approach Integrating Decision Tree, TF-IDF, and BERTopic.
[21] Awotunde, J. B., Adeniyi, E. A., Ogundokun, R. O., & Ayo, F. E. (2021). Application of big data with fintech in financial services. In Fintech with artificial intelligence, big data, and blockchain (pp. 107-132). Singapore: Springer Singapore.
[22] Z. Ren, "Enhancing Seq2Seq Models for Role-Oriented Dialogue Summary Generation Through Adaptive Feature Weighting and Dynamic Statistical Conditioninge," 2024 6th International Conference on Communications, Information System and Computer Engineering (CISCE), Guangzhou, China, 2024, pp. 497-501, doi: 10.1109/CISCE62493.2024.10653360.
[23] Chen, J., Lin, Q., & Allebach, J. P. (2020). Deep learning for printed mottle defect grading. Electronic Imaging, 32, 1-9.
[24] Xie, Y., Li, Z., Yin, Y., Wei, Z., Xu, G., & Luo, Y. (2024). Advancing Legal Citation Text Classification A Conv1D-Based Approach for Multi-Class Classification. Journal of Theory and Practice of Engineering Science, 4(02), 15–22. https://doi.org/10.53469/jtpes.2024.04(02).03
[25] Jurafsky, D. , & Martin, J. H. . (2007). Speech and language processing: an introduction to speech recognition, computational linguistics and natural language processing. Prentice Hall PTR.

[26] Shen, Z., Ma, Y., & Shen, J. (2024). A Dynamic Resource Allocation Strategy for Cloud-Native Applications Leveraging Markov Properties. International Journal of Advance in Applied Science Research, 3, 99-107.
[27] Tekaya, B., Feki, S. E., Tekaya, T., & Masri, H. (2020, October). Recent applications of big data in finance. In Proceedings of the 2nd International Conference on Digital Tools & Uses Congress (pp. 1-6).
[28] Chen, H., & Bian, J. (2019, February). Streaming media live broadcast system based on MSE. In Journal of Physics: Conference Series (Vol. 1168, No. 3, p. 032071). IOP Publishing.
[29] Chen, T., Lian, J., & Sun, B. (2024). An Exploration of the Development of Computerized Data Mining Techniques and Their Application. International Journal of Computer Science and Information Technology, 3(1), 206-212.
[30] VenkateswaraRao, M., Vellela, S., Reddy, V., Vullam, N., Sk, K. B., & Roja, D. (2023, March). Credit Investigation and Comprehensive Risk Management System based Big Data Analytics in Commercial Banking. In 2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS) (Vol. 1, pp. 2387-2391). IEEE.
[31] Teller, & Virginia. (2000). Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition daniel jurafsky and james h. martin (university of colorado, boulder) upper saddle river, nj: prentice hall (prentice hall ser. Computational Linguistics, 26(4), 638-641.
[32] Chen, J., Zhang, X., Wu, Y., Ghosh, S., Natarajan, P., Chang, S. F., & Allebach, J. (2022). One-stage object referring with gaze estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 5021-5030).
[33] Wu, Z. (2024). Large Language Model Based Semantic Parsing for Intelligent Database Query Engine. Journal of Computer and Communications, 12(10), 1-13.
[34] Shakya, S., & Smys, S. (2021). Big data analytics for improved risk management and customer segregation in banking applications. Journal of IoT in Social, Mobile, Analytics, and Cloud, 3(3), 235-249.
[35] Liang, X., & Chen, H. (2019, August). HDSO: A High-Performance Dynamic Service Orchestration Algorithm in Hybrid NFV Networks. In 2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS) (pp. 782-787). IEEE.
[36] Murugan, M. S. (2023). Large-scale data-driven financial risk management & analysis using machine learning strategies. Measurement: Sensors, 27, 100756.
[37] Hasan, M. M., Popp, J., & Oláh, J. (2020). Current landscape and influence of big data on finance. Journal of Big Data, 7(1), 21.