The results confirm the effectiveness of combining transformer-based machine learning models, psychological lexicons, and contextual information within a single architecture. The developed system can be applied in customer support services, intelligent virtual assistants, educational platforms, and other dialogue-based applications. Future research may focus on multilingual emotion recognition, sarcasm detection, and further improvement of context-aware processing mechanisms.

REFERENCES

1. Demszky, D., Movshovitz-Attias, D., Ko, J., Cowen, A., Nemade, G., & Ravi, S. (2020). GoEmotions: A dataset of fine-grained emotions. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (pp. 4040–4054). https://doi.org/10.18653/v1/2020.acl-main.372
2. Mohammad, S. M., & Turney, P. D. (2013). Crowdsourcing a word-emotion association lexicon. Computational Intelligence, 29(3), 436–465. https://doi.org/10.1111/j.1467-8640.2012.00460.x
3. Staiano, J., & Guerini, M. (2014). DepecheMood: A lexicon for emotion analysis from crowd-annotated news. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, 427–433. DOI:10.3115/v1/P14-2070
4. Zhong, P., Wang, D., & Miao, C. (2019). Knowledge-enriched transformer for emotion detection in textual conversations. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, 165–176. https://doi.org/10.18653/v1/D19-1016

Inna Bundiuk
Kyiv National University of Technologies and Design
(Kyiv, Ukraine)
Maryna Vyshnevska
Kyiv National University of Technologies and Design
(Kyiv, Ukraine)

THE ROLE OF OPERATING SYSTEMS IN MODERN DEVICES

In today’s world of rapid digitalization, global connectivity, and ever-increasing computing power, operating systems (OS) hold a crucial position. With the rise of mobile technologies, cloud computing, and the Internet of Things (IoT), there is a growing need for advanced software capable of coordinating hardware resources effectively. The OS is no longer a passive background utility—it has become the

fundamental layer that shapes performance, security, and user experience across modern devices.

One of the core responsibilities of an operating system is process management, which involves overseeing the entire lifecycle of a process—from its creation, through execution, to its eventual termination. The operating system ensures that each process receives the necessary computational resources, particularly CPU time, to perform its tasks efficiently and without conflict. To manage multiple processes simultaneously, modern operating systems implement various process scheduling algorithms. These algorithms are crucial in determining the order in which processes are assigned to the CPU, with the objective of optimizing performance metrics such as throughput, CPU utilization, waiting time, and response time (Rawat, 2025, 8).

The evolution of operating systems has passed through several defining stages. In the early phases (1945–1980), there was a transition from the absence of an OS to the creation of the first batch processing systems, and later to multitasking and multi-user systems. During this period, operating systems evolved from manual control to autonomous tools for scheduling and resource management, laying the foundations of modern OS architecture.

The fourth stage, which continues to this day, represents a qualitative breakthrough in OS development, encompassing personal computers, mobile devices, server systems, and cloud computing. Operating systems of this period provide multitasking, network interaction, interactive operation, high levels of security, and support for diverse hardware platforms. They have become universal tools for both everyday users and critical scientific and engineering applications. Windows, macOS, and Linux have shaped the modern OS market, each demonstrating different approaches to architecture, interface, and distribution models, while their evolution trends toward integration with the network, cloud services, and the mobile environment (Humennyy ,2025, 1293-1294).

The mobile and digital content industries highlight the adaptability of operating systems. The emergence of Android and iOS introduced new design priorities: energy efficiency, touch-based interfaces, and constant connectivity. Unlike desktop systems, mobile OS must balance performance with battery life and thermal limits. At the same time, virtualization and cloud computing have expanded the OS's role, allowing a single physical server to host multiple virtual environments and optimize resource use in global data centers. This shift toward "software-defined" infrastructure means the OS now manages virtual CPUs and memory as if they were physical components.

Security and data protection remain among the greatest challenges for OS developers. As devices become increasingly interconnected, the OS serves as a gatekeeper, defending sensitive information against cyber threats. This involves access control, encryption, secure kernel design, and sandboxing. While early systems lacked strong security, modern operating systems must adopt "security-by-design" principles to counter sophisticated malware and breaches. The growing complexity of software enlarges the attack surface, requiring constant updates, hardware-level protections such as TPM modules, and evolving defense mechanisms.

The security subsystem of an operating system is designed to ensure data protection, user authentication, and access control. It safeguards system integrity by preventing unauthorized access to system resources, files, and processes. The OS enforces authentication mechanisms (such as usernames and passwords), manages user permissions, and implements encryption and auditing features to track and control system activity. For example, creating and enforcing strong passwords is a fundamental method of authentication used to verify user identity before granting access. By implementing these security features, the OS plays a vital role in maintaining confidentiality, integrity, and availability of data in both personal and enterprise computing environments (Rawat, 2025, 11).

Another factor in the success of any OS is its support for a wide ecosystem of hardware and software. Device drivers and standardized APIs allow developers to build

applications without needing deep knowledge of the underlying electronics. The OS also ensures smooth networking, handling complex protocols so that data travels reliably across the global internet. With the advent of 5G and beyond, operating systems must manage higher throughput and lower latency, reinforcing their role as the backbone of modern communication.

In conclusion, operating systems today perform a wide range of roles—from managing low-level hardware to enabling high-level user interaction. Trends such as cloud-native applications and quantum computing promise to expand these responsibilities even further. The future of computing depends on the OS continuing to evolve as a secure, stable, and intelligent bridge between human intent and machine execution. For professionals in the modern technological landscape, practical experience with these systems and a deep understanding of their inner workings are essential.

REFERENCES:

1. Clausing, L., & Platzner, M. (2022, May). ReconOS 64: A hardware operating system for modern platform FPGAs with 64-Bit support. In 2022 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW) (pp. 120-127). IEEE.
2. Hansen, P. B. (1973). Operating system principles. Prentice-Hall, Inc.
3. Humennyi, P.V., Fedorchuk, A.L., & Los, M.V. (2025). Evoliutsiia operatsiinykh system: vid paketnoi obrobky danykh do virtualizovanykh seredovyshch. [Evolution of operating systems: From batch data processing to virtualized environments]. Nauka i tekhnika sohodni, No. 49, pp. 1281–1296. [in Ukrainian].
4. Rawat, V. (2025). A brief study on Operating Systems along with their history, types, working, challenges and advancements. DOI: https://doi. org/10.5281/zenodo, 15869774.