

4. Shihao Jiangb , Yu Zhanga , Junqiang Lia , Hongfang Yua , Long Luo a , Gang Suna. A Survey of Network Protocol Fuzzing: Model, Techniques and Directions. URL: <https://arxiv.org/pdf/2402.17394> (Accessed: 11 November 2025).
5. Andrea Arcuri, Man Zhang, Susruthan Seran, Juan Pablo Galeotti, Amid Golmohammadi, Onur Duman, Agustina Aldasoro, Hernan Ghianni. Tool report: EvoMaster - black and white box search-based fuzzing for REST, GraphQL and RPC APIs. Automated Software Engineering (Autom. Softw. Eng.), Vol. 32, Article 4. 29 November 2024.

СЕКЦІЯ 2. ПРОГРАМНА ІНЖЕНЕРІЯ

РОЗПІЗНАВАННЯ ЕМОЦІЙНИХ СТАНІВ ПІД ЧАС ПЕРЕГЛЯДУ ВІДЕО НА ОСНОВІ ДАНИХ GSR, ПУЛЬСУ ТА ВІЗУАЛЬНОГО ВІДСТЕЖЕННЯ

Глибовець А.М., Хмель С.М. / Hlybovets A., Khmel S.

Національний університет "Києво-Могилянська академія" / National University of Kyiv-Mohyla Academy

04655, Київ, вул. Григорія Сковороди, 2, факультет інформатики

E-mail: a.glybovets@ukma.edu.ua, svitlana.khmel@ukma.edu.ua

This research focused on developing a prototype system for recognizing users' emotional states during video viewing through the integration of biometric and behavioural data. The system combines galvanic skin response (GSR), heart rate monitoring, and gaze tracking using the webgazer.js library to assess emotional arousal and engagement in real time. Biometric signals are collected via Arduino from MAX30102 and GSR Grove sensors and transmitted through the DSD TECH HM-10 Bluetooth Low Energy (BLE) module. A custom Chrome extension detects active video playback, tracks gaze coordinates, and synchronizes visual and physiological indicators into a unified JSON data structure. Experimental results confirm the system's feasibility for analysing user emotions and suggest potential applications in education, neuromarketing, and human-computer interaction.

Keywords: biometric reactions, emotional state, Arduino, MAX30102, GSR Grove, BLE, webgazer.js, human-computer interaction, BCI.

У сучасну епоху активної цифровізації зростає інтерес до систем, здатних розпізнавати емоційні стани людини у реальному часі. Такі системи мають важливе значення для розвитку людино-комп'ютерної взаємодії, адаптивних навчальних середовищ, маркетингових досліджень і нейроінтерфейсів. Особливо перспективним є підхід, що поєднує біометричні сенсори та аналітику поведінки користувача під час споживання мультимедійного контенту.

Метою даної роботи є створення прототипу системи розпізнавання емоційних станів користувача під час перегляду відео, який об'єднує дані гальванічної реакції шкіри (GSR), частоти серцевих скорочень (ЧСС) та напрямку погляду. Основою реалізації є мікроконтролерна платформа Arduino Uno, що забезпечує збір фізіологічних показників у режимі реального часу. Для вимірювання пульсу та рівня насиченості крові киснем використано сенсор MAX30102, а для фіксації електропровідності шкіри — модуль GSR Grove. Передача даних здійснюється через модуль DSD TECH HM-10 за технологією Bluetooth Low Energy (BLE), що забезпечує бездротову взаємодію між апаратною та програмною частинами системи.

Програмна складова реалізована у вигляді розширення для браузера Google Chrome, яке взаємодіє з апаратною частиною, приймає біометричні сигнали та синхронізує їх із поведінковими показниками користувача. Розширення автоматично виявляє активне відео на сторінці YouTube, відстежує його стан (позицію, тривалість, паузу, повноекранний режим), а також за допомогою бібліотеки webgazer.js визначає координати погляду користувача на екрані. Отримані біометричні та візуальні дані об'єднуються в уніфікований формат і зберігаються у вигляді структурованого файлу JSON. Такий підхід дозволяє формувати часові ряди емоційних та фізіологічних змін під час перегляду контенту.

Програмна складова складається з таких компонентів:

9. **Gaze Tracking:** відображає інтерфейс калібрування, визначає координати погляду користувача; виводить візуальний індикатор напрямку погляду; слухач `webgazer.setGazeListener` передає координати у `trackingData.updateGazeData`.
10. **Data Receiver:** підключається до BLE-модуля Arduino; приймає та розпізнає вхідні дані; синхронізує фізіологічні показники з координатами погляду.
11. **UI/UX Module:** кнопки запуску/зупинки відстеження, повідомлення про стан з'єднання.
12. **Messaging System:** забезпечує обмін даними між фоновим процесом, `popup`-вікном розширення та вкладкою браузера;
13. **Інтеграція та потік даних;** Arduino Uno зчитує дані з MAX30102 та GSR-сенсора, дані передаються через HM-10 BLE до комп'ютера користувача; Chrome Extension приймає ці дані та синхронізує з координатами погляду, які надходять від `webgazer.js`; інтегровані дані доступні для зберігання та подальшого аналізу.

Розроблена система побудована за модульним принципом. Кожен компонент — сенсори, модуль зв'язку, програмний інтерфейс та система обробки даних — функціонує автономно, що спрощує оновлення та розширення функціональності. Архітектура системи забезпечує масштабованість: можливе додавання нових типів сенсорів, інтеграція алгоритмів машинного навчання для автоматичної класифікації емоцій чи підключення альтернативних платформ, наприклад Raspberry Pi.

Під час експериментів було підтверджено працездатність прототипу. Система успішно зчитує та передає показники GSR, пульсу і SpO₂, синхронізує їх із координатами погляду та станом відео. У результаті формується повна картина фізіологічних реакцій користувача у процесі сприйняття медіаконтенту. Отримані дані можуть бути використані для побудови моделей оцінки емоційної залученості, візуалізації біологічного зворотного зв'язку або розроблення персоналізованих систем навчання.

Разом з тим, під час тестування виявлено низку обмежень. Зокрема, передача даних через BLE супроводжується затримками, що впливають на синхронізацію між сенсорами. Сенсор MAX30102 не є медичним приладом і демонструє похибки вимірювання, особливо за змінного освітлення чи неправильного положення пальця. Крім того, бібліотека `webgazer.js` має обмежену точність відстеження погляду, що ускладнює використання системи у неконтрольованих умовах.

Попри зазначені недоліки, розроблений прототип демонструє перспективність поєднання недорогих біометричних сенсорів, бездротової передачі даних та браузерних технологій для створення інтерактивних систем аналізу емоційних станів. Подальші напрями розвитку включають удосконалення механізмів калібрування сенсорів, підвищення стабільності зв'язку, застосування методів фільтрації шумів та використання алгоритмів машинного навчання для автоматизованої класифікації емоцій користувача.

Таким чином, представлена система є гнучкою платформою для інтеграції біологічних і поведінкових даних у процесі людино-комп'ютерної взаємодії. Її модульна архітектура, відкритість і простота реалізації роблять можливим широке застосування у сфері освітніх технологій, психологічних досліджень, медіа-аналітики та розвитку нейрокомп'ютерних інтерфейсів.

Список джерел

- [1] Chrome Developers. *Capabilities: Bluetooth* [Електронний ресурс] // Chrome Developers. – 2025. – Режим доступу: <https://developer.chrome.com/docs/capabilities/bluetooth> (дата звернення: 27.10.2025).
- [2] Instructables. *Guide to Using MAX30102 Heart Rate and Oxygen Sensor* [Електронний ресурс]. – 2025. – Режим доступу: <https://www.instructables.com/Guide-to-Using-MAX30102-Heart-Rate-and-Oxygen-Sens/> (дата звернення: 27.10.2025).
- [3] WebGazer.js. *WebGazer Eye Tracking Library* [Електронний ресурс]. – 2025. – Режим доступу: <https://webgazer.cs.brown.edu/> (дата звернення: 27.10.2025).

[4] Noldus. *What Is Galvanic Skin Response* [Електронний ресурс]. – 2025. – Режим доступу: <https://noldus.com/blog/what-is-galvanic-skin-response> (дата звернення: 27.10.2025).

[5] Google Developers. *Interact with BLE Devices on the Web* [Електронний ресурс]. – 2025. – Режим доступу: <https://developers.google.com/web/updates/2015/07/interact-with-ble-devices-on-the-web> (дата звернення: 28.10.2025).

ПРОБЛЕМИ ФРАГМЕНТАЦІЇ КОНТЕКСТУ В RAG СИСТЕМАХ / PROBLEMS OF CONTEXT FRAGMENTATION IN RAG SYSTEMS

Андрощук М.В. / Androshchuk M.

Національний університет “Києво-Могилянська Академія” / National University of Kyiv-Mohyla Academy

04655, Київ, вул. Григорія Сковороди, 2, факультет інформатики, кафедра інформатики
E-mail: maxym.androshchuk@ukma.edu.ua

This research focuses on context fragmentation in Retrieval-Augmented Generation (RAG) systems, a critical issue where conventional document chunking severs logical connections, leading to poor response quality. It proposes a proactive strategy to manage context by overcoming the conflict between retrieval precision and generation completeness. Key solutions include structured indexing via Semantic Chunking and Parent-Child hierarchies, dynamic methods like Sentence-Window and Auto-Merging retrieval, and post-retrieval optimization using rerankers. A comparative analysis shows that an optimized RAG architecture significantly outperforms Large-Context LLMs (LC-LLMs) by avoiding pitfalls such as the "lost in the middle" problem. The paper concludes that strategic architectural investment in solving fragmentation is a more robust and scalable approach for enterprise applications than simply relying on larger LLM context windows.

Системи Retrieval-Augmented Generation (RAG) є поширеною архітектурною парадигмою, що інтегрує великі мовні моделі (LLM) із зовнішніми базами знань з метою підвищення фактологічної точності та мінімізації генерації необґрунтованої інформації ("галюцинацій")[2]. Функціонування цих систем обмежується основним архітектурним викликом — контекстною фрагментацією. Ця проблема виникає внаслідок процесу розбиття (chunking) вихідних документів на менші сегменти для їх ефективного індексування та векторного пошуку. Таке сегментування може призводити до розриву логічних, семантичних та структурних зв'язків, наявних у початковому документі. У результаті LLM отримує набір розрізнених фрагментів, що знижує когерентність генерованої відповіді, призводить до пропуску ключової інформації та включення нерелевантних даних ("шуму"). Особливо значущою є втрата каузальної цілісності, коли причинно-наслідкові зв'язки виявляються розподіленими між різними чанками, що ускладнює для моделі формування логічно обґрунтованих висновків [3].

Для вирішення цієї проблеми розробляються RAG-архітектури, що переходять від стандартних методів розбиття на частини фіксованого розміру до більш комплексних, контекстно-орієнтованих стратегій[5]. Одним із основних рішень є архітектура Parent Document Retrieval (PDR), яка розв'язує компроміс між вимогами до розміру чанків для пошуку та для генерації. В рамках PDR індексуються малі "дочірні" чанки для забезпечення високої точності семантичного пошуку, тоді як для генерації відповіді LLM надається асоційований з ними більший "батьківський" документ, що дозволяє відновити ширший контекст. Цей підхід часто доповнюється ієрархічним розбиттям, що враховує структурні елементи документа (розділи, параграфи), та гібридними стратегіями вибірки, які поєднують семантичний (векторний) та лексичний (на основі ключових слів) пошук для підвищення надійності вибірки.

Важливою складовою зменшення впливу контекстної фрагментації є постретривальна обробка, зокрема механізми переранжування (Re-ranking) [1]. На початковому етапі Retrieval система прагне забезпечити максимальну повноту (high recall) — тобто знайти якомога більше потенційно релевантних чанків. Проте це часто призводить до великої кількості “шумових” або слабо дотичних фрагментів. Щоб забезпечити якість подальшої генерації, може використовуватися переранжування: спеціальний етап, на якому відібрані фрагменти ранжуються за ступенем релевантності до запиту користувача. Найпоширенішим методом для цього є використання Cross-Encoder моделей, які обробляють запит та документ спільно, як єдину вхідну пару. На відміну від bi-encoder'ів, які оперують лише векторною подібністю між попередньо обчисленими ембедінгами, Cross-Encoder здатні захоплювати більш тонкі смислові взаємозв'язки, синтаксичні залежності та каузальні відношення між словами. Це забезпечує суттєво вищу точність фільтрації.

У практичних системах переранжування часто скорочує початковий набір у десятки або навіть сотні фрагментів до невеликої кількості найбільш релевантних, які потім подаються до