УДК 004.5:004.8

# УЗГОДЖЕНЕ КЕРУВАННЯ ПЕРЕРИВАННЯМ ГОЛОСОВОЇ ВІДПОВІДІ AI-АСИСТЕНТА У ВЕБСИСТЕМІ

Трач В.В.[1]; Попова М.О., к.е.н., доц.[2]
[1,2] Національний університет «Одеська політехніка»
[1,2] Україна, Одеса
[1] 5239204@stud.op.edu.ua; [2] popova.m.o@op.edu.ua

***Анотація.*** *Запропоновано механізм переривання голосової відповіді AI-асистента з узгодженням станів вебклієнта й сервера. Визначено порядок зупинки відтворення, скасування операцій і відкидання застарілих аудіофрагментів. Сформульовано умови коректності та критерії перевірки для вебсистем потокового голосового діалогу.*
***Ключові слова:*** *голосова взаємодія; AI-асистент; потокове оброблення мовлення; переривання від-повіді; узгодження станів.*

**Вступ.** Під час голосової взаємодії з AI-асистентом (асистентом на основі штучного інтелекту) користувач може уточнити запит ще до завершення відповіді. Якщо вебсистема продовжує відтворювати попереднє повідомлення, нова репліка накладається на синтезоване мовлення, а наступна відповідь може спиратися на вже неактуальний контекст. Тому підтримка переривання потребує узгодженого керування записом, генерацією та відтворенням, особливо коли ці операції виконують різні компоненти.

У роботі Moshi запропоновано паралельне моделювання мовлення користувача й системи, що дає змогу враховувати перекриття реплік [1]. Mini-Omni демонструє можливість потокового формування мовленнєвої відповіді в наскрізній моделі [2]. Ці дослідження розвивають модельний рівень голосового діалогу. Для вебсистеми з окремими сервісами розпізнавання, генерації тексту й синтезу мовлення додатково потрібно визначити порядок скасування операцій та оброблення даних, які надходять після переривання. Результати FD-Bench показують, що часті переривання та шум залишаються складними умовами для досліджених голосових систем [3]. Отже, перевірка самого факту зупинки звуку недостатня для оцінювання коректності діалогу.

**Метою дослідження** є підвищення коректності переривання потокової голосової відповіді AI-асистента у вебсистемі шляхом узгодженого керування зупинкою відтворення, скасуванням активних операцій та відкиданням застарілих аудіофрагментів, що запобігає відновленню скасованої відповіді та її змішуванню з наступною.

**Основна частина.** Розглянуто вебсистему, що складається з браузерного клієнта, серверного координатора та готових компонентів автоматичного розпізнавання мовлення, мовної моделі й синтезу голосу. Клієнт записує голос користувача та відтворює аудіо, сервер керує сесією й передає дані між компонентами. У межах дослідження передбачено один прикладний сценарій, одну мову та один синтезований голос; навчання власних моделей не розглядається.

Потокову організацію запропоновано застосовувати насамперед до формування відповіді: завершені текстові фрагменти надходять до синтезатора до закінчення генерації всього повідомлення, а отримане аудіо поступово передається клієнту. Це створює можливість раніше розпочати озвучування, проте фактична затримка залежить від розміру фрагментів, швидкодії сервісів і буферизації. Часткові результати розпізнавання можна

показувати в інтерфейсі, але нову відповідь доцільно формувати після підтвердження завершеної репліки, щоб уникнути генерації за нестійкою транскрипцією.

Для керування відповіддю визначено стани очікування, формування, відтворення та скасування. Стани формування й відтворення можуть перекриватися, оскільки наступні аудіофрагменти створюються під час озвучування попередніх. Запис вхідного мовлення керується окремо й залишається доступним під час відповіді асистента. Сигналом переривання є команда користувача або подія виявлення його мовленнєвої активності. Для голосового запуску потрібне приглушення акустичного відлуння: голос асистента з динаміків не повинен помилково трактуватися як нова репліка.

Кожній відповіді в межах сесії надається ідентифікатор покоління – номер, який клієнт збільшує під час початку нової відповіді або скасування поточної та передає серверу. Текстові фрагменти, запити до синтезатора та аудіоблоки зберігають цей ідентифікатор. Координатор і клієнт приймають дані лише актуального покоління. Номер фрагмента всередині відповіді додатково дає змогу виявляти дублікати та підтримувати порядок відтворення. Таке маркування відокремлює життєві цикли відповідей навіть за асинхронного завершення зовнішніх запитів.

Порядок переривання починається на клієнті. Спочатку поточне покоління позначається неактуальним, після чого зупиняються активні й заплановані джерела звуку та очищується черга відтворення. Лише тоді клієнт надсилає серверу команду скасування із зазначенням сесії та відповіді. Локальна зупинка не очікує серверного підтвердження, тому мережеве запізнення не подовжує відтворення на рівні прикладної логіки. Залишкове звучання в апаратному аудіобуфері враховується окремо під час вимірювань.

Сервер припиняє передавання даних скасованого покоління, очищує його черги та надсилає запити на скасування активної генерації й синтезу, якщо відповідні сервіси підтримують таку операцію. Якщо зовнішній запит неможливо зупинити, його результат відкидається. Перевірка актуальності виконується після завершення кожної асинхронної операції та перед додаванням даних до вихідної черги. Повторна команда скасування тієї самої відповіді не змінює стану нової. Запізнілі повідомлення попереднього покоління також не можуть зменшити актуальний номер.

Необхідність перевірки покоління пояснює такий сценарій: користувач перериває відповідь, клієнт очищує чергу, але після цього завершується раніше розпочатий синтез. Якщо клієнт приймає отриманий аудіоблок без перевірки, черга знову заповнюється й озвучування відновлюється. У запропонованому механізмі цей блок належить скасованому поколінню і відкидається. Отже, очищення черги та заборона приймання застарілих даних виконують різні функції й мають застосовуватися разом.

Для механізму сформульовано три умови коректності: до відтворення допускаються лише блоки актуального покоління; скасована відповідь не переходить назад до відтворення; наступна відповідь не містить аудіоблоків попередньої. За умови перевірки кожного блока ці властивості випливають із незворотного збільшення номера покоління. Це логічне обґрунтування стосується прикладного протоколу й не замінює перевірки реалізації, зокрема браузерного планування звуку та відновлення сесії після втрати з’єднання.

Окремо визначено вимогу до історії діалогу. Згенерований текст не слід автоматично вважати повністю повідомленим користувачу. Після переривання потрібно позначити відповідь як незавершену та, за наявності зіставлення тексту з аудіо, зберегти підтверджений відтворений фрагмент. Інакше асистент може посилатися на пояснення, яке фактично не

прозвучало. За відсутності точного зіставлення до контексту додається ознака переривання без твердження про повне озвучування.

Для подальшої валідації визначено час від завершення репліки користувача до початку звучання відповіді, час від початку переривальної репліки до припинення звучання та частку коректно завершених сценаріїв. Окремо фіксується затримка від виявлення переривання до зупинки, щоб розмежувати роботу детектора та механізму скасування. Для клієнтських інтервалів використовується один монотонний годинник; фактичні межі звучання перевіряються за контрольним записом аудіовиходу.

Порівняння передбачає однакові голосові запити, моделі, налаштування синтезу та мережеві умови. Послідовний режим очікує повного завершення кожного етапу, потоковий передає проміжні фрагменти. В обох режимах застосовується однаковий механізм переривання. Окремим контрольним варіантом є лише зупинка й очищення клієнтської черги без перевірки поколінь. Сценарії охоплюють переривання на початку й у середині озвучування, повторне переривання, запізнілий аудіоблок і відмову сервісу. Для часових показників передбачено повторні прогони з обчисленням медіани та 95-го процентиля; кількісні результати потребують реалізації й експерименту.

**Висновки.** Результатом дослідження є проєкт механізму переривання, у якому локальна зупинка відтворення поєднана із серверним скасуванням операцій та перевіркою актуальності кожного фрагмента. На сценарії запізнілого завершення синтезу показано, чому очищення аудіочерги саме по собі не запобігає відновленню скасованої відповіді. Сформульовано умови коректності обміну, вимогу до збереження незавершеної відповіді в історії та критерії подальшої валідації. Рішення може бути використане під час реалізації вебсистеми голосової взаємодії з готовими AI-компонентами; фактичний виграш у швидкодії має бути встановлений експериментально.

**Використання інструментів ШІ.** ChatGPT (OpenAI, модель GPT-6) використано для мовного редагування анотацій, вступу, основної частини й висновків. ШІ не використовувався для отримання експериментальних даних.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

[1] Défossez, A., Mazaré, L., Orsini, M., Royer, A., Pérez, P., Jégou, H., Grave, E., & Zeghidour, N. (2024). *Moshi: A speech-text foundation model for real-time dialogue* [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2410.00037

[2] Xie, Z., & Wu, C. (2024). *Mini-Omni: Language models can hear, talk while thinking in streaming* [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2408.16725

[3] Peng, Y., Chao, Y.-W., Ng, D., Ma, Y., Ni, C., Ma, B., & Chng, E. S. (2025). *FD-Bench: A full-duplex benchmarking pipeline designed for full duplex spoken dialogue systems* [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2507.19040

Trach V.V., Popova M.O.

**Coordinated interruption control for AI assistant voice responses in a web system**

***Abstract****. A mechanism for interrupting an AI assistant's voice response is proposed through coordinated client and server state management. The sequence of playback termination, operation cancellation and stale audio rejection is defined. Correctness conditions and validation criteria are formulated for streaming voice dialogue web systems.*

***Keywords****: voice interaction; AI assistant; streaming speech processing; response interruption; state coordination.*