### Ювженко Денис Ігорович, Аспірант, Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", Київ, Україна

ORCID: https://orcid.org/0009-0005-8744-226X

Чимшир В’ячеслав Іванович, Аспірант, Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", Київ, Україна  
ORCID: <https://orcid.org/0009-0005-2555-5373>

Володимир Шимкович, Кандидат технічних наук,

Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", Київ, Україна  
ORCID: https://orcid.org/0000-0003-4014-2786

Сергій Федорович Теленик, Доктор технічних наук,

Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", Київ, Україна  
ORCID: <https://orcid.org/0000-0002-1291-0373>

Автоматизація бізнес-процесів за допомогою мультиагентної ШІ-системи

Сьогодні в галузі інформаційних технологій ключовими чинниками впливу на покращення клієнтського досвіду, зниження витрат та оптимізації бізнес-операцій є автоматизація процесів та ефективне використання штучного інтелекту (ШІ) [1]. Одним із найбільш перспективних напрямів є агентні складені ШІ-системи, які дозволяють розподіляти завдання між різними спеціалізованими агентами [2], кожен із яких відповідає за окремий аспект роботи.

У цій роботі представлено прототип агентної ШІ-системи, який демонструє інтеграцію великих мовних моделей — Large Language Models (LLM) у бізнес-процеси діяльності компаній галузі телекомунікацій, яка стає все більш популярною сьогодні [3].  
Як приклад, була побудована система для телеком-провайдера відповідно до стандартів TM Forum [4]. Реалізація базується на LangChain [5] та LangGraph — фреймворках, що дозволяють створювати керовані та структуровані ланцюги запитів між агентами.

Агентні складені ШІ-системи — це багатокомпонентні системи, які складаються з кількох спеціалізованих агентів, кожен із яких виконує певну задачу, а їхня взаємодія координується агентом-оркестратором або супервайзером. До переваг використання такого підходу належать гнучкість, масштабованість та можливість спеціалізації агентів для виконання окремих завдань.

Традиційні ШІ-системи часто функціонують як монолітні моделі, що намагаються розв’язати широкий спектр задач всередині однієї нейромережі, великої мовної чи іншої моделі ШІ. Агентні системи, навпаки, розподіляють відповідальність між незалежними агентами, які можуть взаємодіяти між собою через стандартизовані API або оркестраційні механізми.

У цій роботі пропонується агентна ШІ-система, яка поєднує потужність LLM з генерацією з доповненим пошуком — Retrieval-Augmented Generation (RAG) [6], оркестрацією через фреймворк LangGraph та інтеграцією із системами телеком-оператора. Представлений прототип демонструє здатність системи до опрацювання текстів природньою мовою, міркувань і розв'язання складних проблем шляхом декомпозиції на прикладі одного бізнес-процесу. У перспективі вбачається розширення системи для масштабної інтеграції у інформаційні системи телеком-операторів.

Агентний підхід передбачає розподіл завдань між автономними агентами, кожен із яких має свою спеціалізацію. У запропонованій системі LLM, RAG та компоненти, які реалізують спеціалізовані функції, виконують роль інструментів, якими користуються агенти для виконання складних завдань. Це дозволяє:

* Забезпечити гнучке управління потоками інформації.
* Використовувати найкращі практики інтелектуального оброблення даних.
* Оптимізувати пошук інформації та генерацію відповідей у реальному часі.

Основним елементом управління у складених агентних системах є агент-оркестратор (або супервайзер). Його завдання полягає у тому, щоб декомпозувати складні проблеми на підпроблеми, координувати їх розв’язання, виконання запитів, передавати їх між агентами та забезпечувати правильну послідовність операцій.

Використання фреймворку LangGraph дозволяє реалізувати графову модель управління взаємодією агентів. При застосуванні традиційного підходу до ШІ-систем існує проблема неконтрольованої генерації відповідей моделлю. LangGraph вирішує цю проблему, формуючи правильну послідовність виконання запитів у визначеному відповідними компонентами ШІ-системи порядку.

Ключовою частиною пропонованої ШІ-системи є агент, який реалізує RAG, базуючись на підході, що поєднує генеративні LLM із зовнішніми джерелами знань. RAG-компонент відповідає за отримання та оброблення інформації з бази знань для генерації точних та релевантних відповідей. Це досягається шляхом додавання необхідного контексту у вигляді конкретних даних, які отримані з векторної бази даних [7], відповідно до запиту великої мовної моделі.  
Таким чином, велика мовна модель буде формувати відповідь базуючись на заданому контексті, а не лише на тому обсязі інформації, на якому її було натреновано.  
Це дозволяє значно зменшити ймовірність галюцинацій великих мовних моделей та забезпечити генерацію відповідей на основі актуальних верифікованих даних.  
Це особливо актуально при використанні таких ШІ-систем для підтримки бізнесової діяльності, коли дані, як і контекст запиту, можуть часто змінюватися. Типовими прикладами таких даних у системах телеком-операторів є тарифи, акції, дані абонентів, тощо.

Здатність запропонованого підходу скласти основу для вирішення практичних проблем продемонстровано на прикладі однієї з властивих підтримці бізнес-процесів телекомунікаційної компанії проблем, а саме проблемі регулювання високого навантаження на службу підтримки клієнтів. У цьому випадку типовими викликами для розробника є:

* Велика кількість однотипних запитів (перевірка рахунку, підключення тарифу, усунення несправностей).
* Існуючі IVR-системи та rule-based традиційні чат-боти засновані на сценаріях неефективні для складних запитів.

На основі запропонованого підходу процес оброблення запитів у системі представлений наступним чином:

1. Користувач надсилає запит (наприклад: "Я хочу змінити свій тарифний план на “Успішний”. ").
2. Комунікаційний агент аналізує намір користувача і передає запит Оркестратору.
3. Оркестратор визначає, які агенти мають виконати операцію (RAG для перевірки деталей тарифів або бізнес-агент для взаємодії з білінговою системою).
4. Агент RAG шукає інформацію про доступні тарифи у векторній БД та передає відповідь Оркестратору.
5. Бізнес-агент використовує API телеком-оператора для здійснення змін у системі білінгу.
6. Фінальна відповідь надсилається користувачеві через комунікаційний агент.

Практичні результати та перспективи масштабування. Для простих запитів, таких як перевірка статусу клієнта, інформації про контракт або інших загальнодоступних даних, використовується агент, що працює з RAG-компонентою. Він отримує відповідну інформацію з бази даних, передає її до LLM, яка генерує відповідь для клієнта. Оркестратор керує цим процесом, а комунікаційний агент передає відповідь користувачу.

Результати:

* Система продемонструвала високу точність відповідей та швидкість оброблення запитів, що значно покращило взаємодію з клієнтами.
* Запропонований підхід є масштабованим, оскільки можна додати додаткові бази даних із розширеною інформацією.
* Продемонстровані системою здатності дозволяють використовувати її не лише для клієнтів, а й для працівників компанії, надаючи їм внутрішню інформацію про процеси та послуги. В цьому випадку пропонується розділення RAG-агентів на внутрішнього, для працівників компанії, та зовнішнього, для клієнтів, аби унеможливити доступ до конфіденційних даних ззовні.

### Автоматизація бізнес-процесів через бізнес-агента.

### Система містить бізнес-агента, який взаємодіє з компонентами ІС телеком-компанії через API. Його завданням є формування контрактів для нових клієнтів.

Результати:

* Система продемонструвала здатність до успішної автоматичної підтримки бізнес-процесу формування контрактів з мінімальним залученням персоналу компанії.
* Отримані результати підтверджують можливість масштабування системи шляхом реалізації інших бізнес-функцій телеком-оператора.
* Створення спеціалізованих агентів для окремих бізнес-задач (наприклад, управління платежами, обслуговування клієнтів, технічна підтримка) дозволить не тільки збільшити ефективність діяльності оператора, але й певним чином реструктурувати бізнес-процеси, зробивши їх більш простими, зручними і логічно обгрунтованими.

Таким чином, отримані результати підтверджують ефективність запропонованого підходу та можливість його масштабування для розширеного використання в телекомунікаційній індустрії. Розширення системи планується шляхом розроблення дадодаткових агентів для реалізації інших бізнес-функцій, підтримки інших бізнес-процесів.

Висновки та перспективи подальшого розвитку.

У роботі представлено агентну складену ШІ-систему, яка інтегрує великі мовні моделі, механізм RAG та агентну оркестрацію для автоматизації бізнес-процесів телекомунікаційної компанії. Запропонований підхід дозволяє ефективно поєднувати можливості генеративного ШІ з доступом до актуальної інформації, що значно покращує якість обслуговування клієнтів та автоматизує внутрішні бізнес-операції.  
Практичні результати підтвердили ефективність запропонованого рішення у двох основних сценаріях:

* Оброблення стандартних запитів через RAG-агента дозволяє швидко та точно надавати клієнтам інформацію про їхні контракти, статус рахунку та інші ключові деталі. Такий підхід добре масштабується завдяки можливості інтеграції нових баз даних, що розширює сферу його застосування на внутрішню корпоративну аналітику.
* Автоматизація формування контрактів за допомогою бізнес-агента довела свою ефективність, дозволяючи значно зменшити навантаження на працівників компанії. Цей підхід може бути розширений на інші бізнес-процеси, такі як управління платежами, технічна підтримка та інші аспекти роботи телеком-оператора.

Отримані результати свідчать про те, що використання агентної архітектури в ШІ-системах є ефективним та гнучким підходом до вирішення широкого спектра бізнес-завдань. Впровадження подібних рішень може значно оптимізувати операційні процеси компаній, зменшити витрати та підвищити якість обслуговування клієнтів.

### Перспективи подальших досліджень та розвитку

1. Розширення агентної архітектури для підтримки додаткових бізнес-функцій.
2. Оптимізація продуктивності системи шляхом удосконалення кешування даних для типових однотипних запитів.
3. Впровадження додаткових моделей машинного навчання для прогнозування запитів клієнтів та адаптації сервісів у режимі реального часу.

Таким чином, запропонована система не лише покращує існуючі методи автоматизації бізнес-процесів, але й відкриває перспективи для подальшого розширення та вдосконалення технологій, що працюють на основі агентного ШІ.

Література

1. Рассел С., Норвіг П. Штучний інтелект: сучасний підхід. 4-е вид. – Pearson, 2020. – 1152 с.
2. Карпус Р., Лазарік А. Підкріплювальне навчання та прийняття рішень у мультиагентних системах // Журнал досліджень штучного інтелекту. – 2021. – Т. 72. – С. 105-135.
3. Льюїс М., Фергюсон К. Великі мовні моделі та їх застосування в автоматизації бізнесу // Огляд штучного інтелекту. – 2022. – Т. 39(3). – С. 415-432.
4. TM Forum. TM Forum Open Digital Framework: Standards for Digital Transformation. – TM Forum White Paper. – Лондон: TM Forum, 2023. – 42 с.
5. Шмідт Т., Бауер А. Оркестрація в мультиагентних системах: LangGraph та його застосування // Журнал інтелектуальних систем. – 2021. – Т. 35(4). – С. 627-642.
6. Choi H., Gupta A., Wang J. LangChain: Building Applications with Large Language Models. – Technical Report. – OpenAI, 2023. – 36 с. – Доступно: [https://www.langchain.com](https://www.langchain.com/).
7. Ян З., Чжан П., Лу Ю. ChromaDB: Масштабований векторний пошук для високорозмірного отримання даних // Матеріали конференції ACM SIGMOD. – 2022. – С. 289-300.