

Андрошук М. В., Кириєнко О. В.

## ПОРІВНЯННЯ СЕРВІСІВ ДЛЯ СТВОРЕННЯ ПИТАЛЬНО-ВІДПОВІДАЛЬНИХ СИСТЕМ

*У статті розглянуто актуальність питально-відповідальних систем, сервісів для створення питально-відповідальних систем Dialogflow, IBM Watson Assistant, Microsoft QnA Maker, LUIS. Наведено вимоги для роботи та особливості створення питально-відповідальної системи в кожному сервісі. Робота може бути цікавою для дослідників у галузі питально-відповідальних систем і хмарних сервісів.*

**Ключові слова:** питально-відповідальні системи, Dialogflow, Microsoft Bot Service, Microsoft QnA Maker, LUIS, IBM Watson Assistant.

### Вступ

Питально-відповідальні системи є важливою частиною сьогодення. Вони відіграють важливу роль у взаємодії з людьми, даючи змогу автоматизувати частину взаємодії цього процесу за допомогою систем. Зокрема, поширеними стали рішення у сфері клієнтської підтримки, рекомендацій та замовлень.

Однак сьогодні існує багато підходів до реалізації питально-відповідальних систем, є різні технології та сервіси для створення таких систем. Порівняння є важливим для вибору рішення, що найкраще відповідатиме вимогам завдань.

Питально-відповідальна система – система, що автоматично відповідає на запитання, поставлені людьми природною мовою, використовуючи попередньо структуровану базу даних або колекцію документів природною мовою [2].

Питально-відповідальні системи можна розглядати як розвиток інформаційного пошуку. Щодо таких систем існує зацікавленість, їх кількість постійно зростає [2], оскільки постійно збільшується обсяг інформації, доступний у мережі Інтернет, і потрібні інструменти пошуку для виокремлення корисної інформації.

Такі системи зазвичай використовують як альтернативу збіркам запитань, що їх часто ставлять. Іншим варіантом використання є клієнтська підтримка, коли користувач спілкується з системою, а не живою людиною.

### Dialogflow

Dialogflow – платформа для взаємодії користувачів із додатками, сайтами, пристроями, ботами та іншими застосуваннями. Декларує під-

тримку оброблення природної мови, можливість оброблення текстових і аудіозапитів та текстові або згенеровані аудіовідповіді [3].

Робота з Dialogflow базується на таких основних концептах: агент (agent), інтент (intent), сутність (entity), контекст (context). Вони є основою для побудови взаємодії з користувачами.

Агент у Dialogflow є віртуальним агентом, що обробляє бесіду з користувачами системи. Він є модулем, який розуміє природну мову та структурує дані, отримані в рамках бесіди для подальшого оброблення [3]. Інтент класифікує намір користувача для ведення бесіди. Кожен агент може мати багато інтенів, які можуть бути використані при взаємодії з користувачем. Кожна репліка користувача зіставляється з наявними інтенетами, і обирається найкраща [3].

Інтент містить тренувальні фрази, що визначають, які репліки користувача викликають цей інтент. Кожен інтент може містити багато тренувальних фраз. При цьому система декларує розширення вказаного списку схожими фразами за рахунок машинного навчання. Загальною рекомендацією є використання не менше ніж 10 тренувальних фраз [3].

Якщо репліка користувача збігається з якимось інтенетом, Dialogflow витягує параметри з репліки. Кожен такий параметр має тип сутності, який визначає, як дані витягатимуться. Для підтримки такого витягання на тренувальних фразах потрібно розмістити параметри і вказати їх тип сутності. Отримані параметри можна використати для побудови відповіді та передання власному сервісу за допомогою webhook [3].

Кожен інтент потребує відповіді. Зокрема, Dialogflow підтримує статичні й динамічні відповіді. Статичні відповіді вказуються в інтенеті,

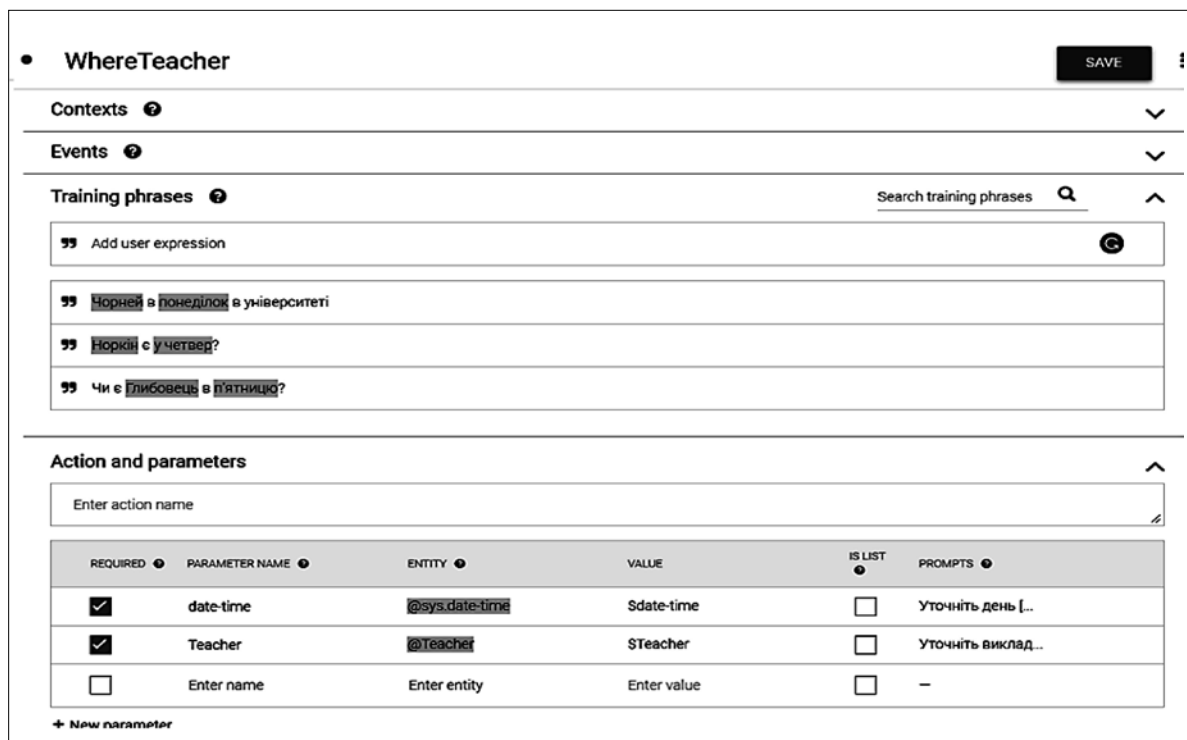


Рис. 1. Частина інтерфейсу Dialogflow

їх може бути декілька для різноманітності відповідей. Для динамічних відповідей є можливість додати параметри, отримані з репліки користувача, або використання власного сервісу зі створення там відповіді. Базовою відповіддю є текст, але інші типи (картинка, звук тощо) також можливі залежно від платформи, де працюватиме агент. При цьому відповідь може містити уточнювальне запитання, якщо не всі дані отримані в рамках одного інтену [3].

Сутність – це параметр, який може бути у репліці користувача. Кожна сутність має тип. Dialogflow має такі початкові типи: час, дата, довжини та ін. При цьому є можливість визначити власний тип, якщо готові не підходять. Кожному типу сутності відповідають конкретні сутності, які потрібно вказати. Декілька різних слів можуть відповідати одній сутності [3].

Контексти в Dialogflow близькі до контекстів у природній мові. Вони потрібні для побудови сценаріїв діалогу. Зокрема, кожен інтену може приймати і повертати контексти. При активації інтену повертається вихідний контекст. І при наступному виборі інтену вибиратимуться ті, у яких вхідний контекст збігається з попереднім вихідним [3].

У разі потреби динамічних відповідей Dialogflow дає змогу додати інтеграції зі стороннім сервісом. Кожен інтену має налаштування, що вказує потребу в зовнішньому сервісі. Якщо воно

не увімкнене, то буде використано статичну відповідь. Тому для динамічних відповідей потрібно, щоб це поле було увімкнено [3].

Для початку роботи з Dialogflow потрібно в Google-акаунті перейти до консолі сервісу (<https://dialogflow.cloud.google.com/>). Після цього – створити агента, де вказати ім'я агента, задану мову і часовий пояс, за яким запити вказуватимуть час. Далі потрібно вибрати, на яких платформах працюватиме агент, і виконати кроки для їх додавання. Потім необхідно визначити інтенти та інші параметри агента й запустити тренування агента.

### IBM Watson Assistant

IBM Watson Assistant – платформа для побудови питально-відповідальних систем. Створена компанією IBM у рамках проекту комп'ютерного штучного інтелекту IBM Watson [5].

Основні концепти, якими оперує IBM Watson Assistant, – інтенти та сутності. Інтену є колекцією реплік користувачів з однаковим значенням. Якщо у репліці користувача розпізнає інтену, то він піде за сценарієм діалогу [5]. Сутність представляє інформацію із репліки користувача, яка стосується його наміру [5].

Підтримується інтеграція з власним сайтом, через окрему вебсторінку, Facebook Messenger, Slack, Intercom. Також є можливість створити власну інтеграцію за допомогою API [5].

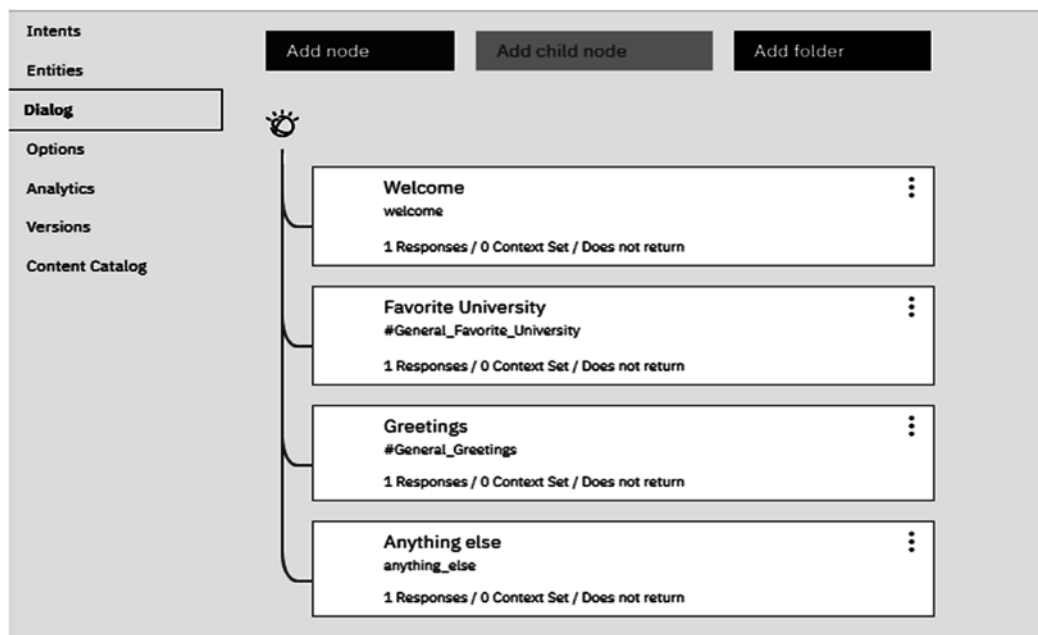


Рис. 2.  
Частина  
інтерфейсу  
IBM Watson  
Assistant

Для початку роботи потрібно мати IBM Cloud акаунт. Далі: вибрати створення Watson Assistant; вибрати регіон, де розміщуватиметься сервіс, вказати тарифний план та вибрати назву; натиснути на кнопку створення. Після цього відкриється вікно налаштувань сервісу.

Потрібно вказати ім'я діалогу, чи має пропонуватися користувачеві перехід на цей діалог, умова запуску діалогу, відповідь користувачеві та дія після надання відповіді. Після надання відповіді можливе звичайне очікування наступної репліки або перехід на інший діалог. Перехід може бути з очікуванням репліки, відповіддю за відповідності попередньої репліки та безумовною відповіддю.

### Сервіси Microsoft Azure

Microsoft Azure – платформа хмарних обчислень, створена компанією Microsoft. Пропонує SaaS, PaaS, IaaS рішення. Для створення питально-відповідальної системи має декілька сервісів: Personalizer – для персоналізованих відповідей користувачу, QnA Maker – база знань для відповідей, Language Understanding – для виділення намірів у репліках користувача, Text Analytics – для виділення настрою, ключових фраз та іменованих сутностей, Azure Bot – для взаємодії з користувачем та викликів інших сервісів [6].

Language Understanding (LUIS) – хмарний сервіс, що надає API для оброблення реплік користувача. Використовує машинне навчання для визначення значення та отримання релевантної

детальної інформації із фраз користувача. Будь-який розмовний додаток, що працює з природною мовою, може використовувати LUIS, наприклад чат-боти. Сервіс підтримує англійську, китайську, французьку, італійську, німецьку, іспанську, японську мови та деякі інші, але української серед них немає [1].

Додаток-клієнт відправляє текст, уведений користувачем, на ендпоінт LUIS. Після чого сервіс зіставляє текст із можливими інтентами і вибирає найбільш імовірний. Далі намагається виділити сутності, наявні в повідомленні, а потім передає ці дані додатку-клієнту для їх подальшого оброблення [1].

QnA Maker – хмарний NLP сервіс, який створює шар природної мови над даними. Може використовуватися для найбільш доречної відповіді користувачеві з бази знань при запиті природною мовою. Користувачами сервісу є розмовні додатки, які взаємодіють із користувачами природною мовою [1].

QnA Maker рекомендують використовувати у разі наявності статичної інформації у базі знань для відповідей або за однакової відповіді різним користувачам на однакові запитання. Прикладами такої інформації є FAQ, документація, брошури, статична інформація, наприклад pdf-документи, вебресурси з відповідями тощо [1].

Додаток-клієнт відправляє запитання на ендпоінт QnA Maker. Після цього сервіс, використовуючи базу знань, надає відповідь і можливі уточнення для видавання найкращої відповіді. Додаток-клієнт отримує відповідь, надану сервісом,

з оцінкою відповідності й може перенаправити її користувачеві або запропонувати уточнення [1].

Для початку роботи потрібно мати акаунт Microsoft, кошти на рахунку Azure для розміщення сервісів та увійти у Microsoft Azure.

Основою для роботи є Azure Bot Service Bot. Цей сервіс безпосередньо взаємодіє з користувачами, отримує та відправляє повідомлення, наприклад чат-боти [1].

Далі потрібно обрати створення нового бота, після чого сервіс пропонує обрати тип бота: Web App Bot або Bot Channels Registration. Перший варіант передбачає, що бот буде створений та розгорнутий всередині Azure як Azure App Service Web App. Другий передбачає наявність бота, розгорнутого поза Azure, та його під'єднання до Bot Service, цей варіант у рамках цієї роботи не розглядається.

Після цього можна перевірити поведінку бота у інтерфейсі Azure інструментом Web Chat, що доступний у консолі сторінки бота. Оскільки базова версія бота не має достатніх можливостей для створення власної системи, то потрібно завантажити код бота й розпакувати в окрему директорію. Можливе використання шаблонів для ботів, які розміщені в інтернеті.

Перед початком роботи потрібно створити LUIS сервіс, зазначивши назву, мову, групу авторів, і ключ.

Після створення – додати інтенти, які мають розпізнавати сервіс. Є стандартні інтенти для погоди, автоматизації будинку, замовлення квитків та ін. Якщо ці інтенти не підходять, потрібно створити нові й додати відповідні фрази і сутності, що можуть бути передані. Створимо два інтенти – FavoriteUniversity та DeanComputerScience.

Далі необхідно запустити тренування моделі та опублікувати сервіс. Інтерфейс LUIS дає можливість відразу перевірити модель. Користувач надсилає повідомлення, а у відповідь йому повертається інтент з оцінкою ймовірності.

Наступним сервісом є QnA Maker. Потрібно створити нову базу знань, де вказати параметри розміщення сервісу в Azure, мову бази знань, її назву та документи для її наповнення.

Після наповнення бази знань можна перевірити і доповнити іншими парами запитання – відповідь. Для кожної такої пари можна додати підказку для подальших запитань.

Після наповнення потрібно опублікувати сервіс для повноцінного використання.

Робота з ботом передбачає наявність директорії вихідних кодів сервісу. У директорії містяться файли, які відповідають за налаштування бота та виклики інших сервісів. Файл .env містить ідентифікатори, ключі та адреси про сам сервіс бота та сторонні сервіси, що їх використовуватиме бот, у цьому випадку LUIS і QnA Maker.

Щоб перевірити роботу бота, можна розгорнути його на власній машині або опублікувати на Azure. Для локальної перевірки потрібно запустити бота командою npm start і використати Bot Framework Emulator – інструмент від Microsoft для локальної перевірки поведінки ботів. За замовчуванням бот стає доступний за адресою http://localhost:3978/api/messages. При відкритті бота необхідно вказати дані сервісу, який використовує бот: MicrosoftAppId та MicrosoftAppPassword. В іншому разі запити до сторонніх сервісів будуть невдалими через потребу в авторизації.

Для розгортання бота в Azure потрібно встановити Azure CLI та увійти у свій обліковий

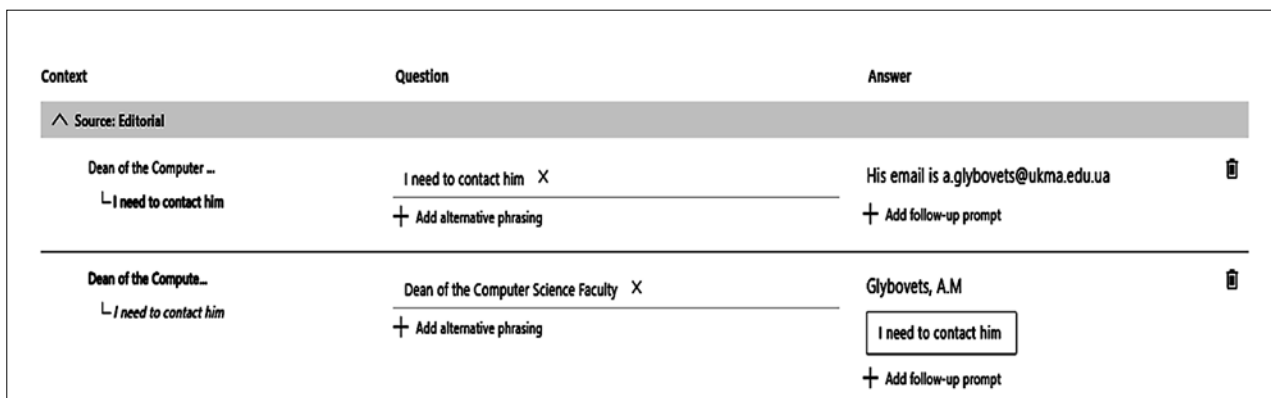


Рис. 3. Частина інтерфейсу Microsoft QnA Maker

запис через команду `az login`. Далі архівувати директорію, де розташовані вихідні коди бота, й виконати команду `az webapp deployment source config-zip`, вказавши в параметрах групу ресурсів, назву сервісу та шлях до zip-архіву. Через декілька хвилин бот буде розгорнуто в Azure з можливістю взаємодії через інтернет.

### Висновки

Було розглянуто сервіси Dialogflow, IBM Watson Assistant, Microsoft QnA Maker та Microsoft LUIS. Найвні рішення дають змогу будувати рішення різного рівня складності. Кожен сервіс має свої переваги й недоліки. Важливим для вибору є задачі, які повинна вирішувати створювана питально-відповідальна система, та обсяги використання системи.

Кожен сервіс має обмеження мов, які підтримуються. Dialogflow підтримує 19 мов, IBM Watson Assistant – лише 10, Microsoft QnA Maker – 52,

а LUIS – лише 17. Усі сервіси підтримують англійську мову, але лише Dialogflow і Microsoft QnA Maker підтримують українську та російську мови, решта сервісів не надає можливості працювати з ними.

Сервіси мають можливості інтеграції з месенджерами. Зокрема, у всіх сервісів є підтримка Slack і Facebook Messenger. Решта має різну підтримку у різних сервісів. Всі сервіси надають можливість створити власну інтеграцію, якщо сервіс їх не надає, через відповідні API.

Ціни залежать від кількості запитів на місяць із можливим обмеженням за кількістю на хвилину. Кожен сервіс за певних умов пропонує безкоштовний план, однак сервіси Microsoft вимагатимуть оплати потужностей Azure.

У перспективі можливий огляд інших сервісів, таких як Amazon Lex, і вимірювання точності та повноти відповідей систем, створених різними сервісами на однакових даних.

### Список літератури

1. Azure Cognitive Services documentation [Electronic resource]. – Mode of access: <https://docs.microsoft.com/en-us/azure/cognitive-services>.
2. Calijorne Soares M. A literature review on question answering techniques, paradigms and systems / M. Calijorne Soares, F. Parreiras // Journal of King Saud University – Computer and Information Sciences. – 2018.
3. Dialogflow documentation [Electronic resource]. – Mode of access: <https://cloud.google.com/dialogflow/docs>.
4. Dialogflow [Electronic resource]. – Mode of access: <https://en.wikipedia.org/wiki/Dialogflow>.
5. IBM Cloud Docs [Electronic resource]. – Mode of access: <https://cloud.ibm.com/docs/assistant>.
6. Microsoft Azure [Electronic resource]. – Mode of access: [https://en.wikipedia.org/wiki/Microsoft\\_Azure](https://en.wikipedia.org/wiki/Microsoft_Azure).

### References

- Calijorne Soares, M., & Parreiras, F. (2018). A literature review on question answering techniques, paradigms and systems. *Journal Of King Saud University – Computer And Information Sciences*. <https://doi.org/10.1016/j.jksuci.2018.08.005>
- Dialogflow. (2020). In *Wikipedia*. Retrieved from <https://en.wikipedia.org/wiki/Dialogflow>.
- Google. (n. d.). Dialogflow documentation | Google Cloud. *Dialogflow documentation*. Retrieved from <https://cloud.google.com/dialogflow/docs>.
- IBM. (2020). IBM Cloud Docs. *Getting started with Watson Assistant*. Retrieved from <https://cloud.ibm.com/docs/assistant>.
- Microsoft Azure. (2020). In *Wikipedia*. Retrieved from [https://en.wikipedia.org/wiki/Microsoft\\_Azure](https://en.wikipedia.org/wiki/Microsoft_Azure).
- Microsoft. (2020). Azure Cognitive Services documentation | Microsoft Docs. *Azure Cognitive Services documentation*. Retrieved from <https://docs.microsoft.com/en-us/azure/cognitive-services>.

*M. Androshchuk, O. Kyriienko*

## COMPARISON OF SERVICES FOR CREATING QUESTION-ANSWERING SYSTEMS

*Question-answering systems are widely used nowadays. They allow automating interaction between people. Common solutions include client support, recommendations, and order processing.*

*A question-answering system is a system that answers natural language questions using a structured database or natural language document collection. It can be considered as the development of information retrieval.*

*Three are different services for creating question-answering systems. In this work selected Dialogflow, IBM Watson Assistant, Microsoft QnA Maker, and LUIS. Each service has its advantages and disadvantages.*

*The main concepts of Dialogflow are agents, intents, entities, context. An agent is a virtual agent with a natural language understanding that handles conversations with users. An intent represents user intention for conversation turn. Each agent has its intents. Entities are parameters of intent that Dialogflow extracts from user expressions. A context in Dialogflow similar to natural language context and helps to control the flow of conversation.*

*The main concepts of IBM Watson Assistant are intents and entities. Intents represent users' goals. User inputs analyzed to select the most appropriate intent and choose dialog flow. An entity represents information in intent that is relevant to the user's goal.*

*Microsoft QnA Maker uses a concept of a knowledge base that is filled with information. Requests from the user are analyzed to extract the most appropriate information from the knowledge base. LUIS uses concepts of intent and entities. LUIS uses schemas to extract intent or entity from an input.*

*Important for the choice are the tasks to be solved by the created question-answer system and the volume of system use. Each service has restrictions on the languages that are supported. Dialogflow supports 19 languages, IBM Watson Assistant only 10, Microsoft QnA Maker 52 languages, but LUIS only 17. All services support English, but only Dialogflow and Microsoft QnA Maker support Ukrainian and Russian. Each service has standard integrations with messengers. All services support Slack and Facebook Messenger. The rest have different support from different services. All services provide options to create a custom integration for non-standard systems if the service does not provide them, through the appropriate APIs.*

*Prices depend on the number of requests per month with a possible limit on the number per minute. Each service offers a free plan under certain conditions, but Microsoft services will require payment for Azure capacity.*

**Keywords:** question-answering systems, Dialogflow, Microsoft Bot Service, Microsoft QnA Maker, LUIS, IBM Watson Assistant.

*Матеріал надійшов 09.06.2020*



Creative Commons Attribution 4.0 International License (CC BY 4.0)