

АНАЛІЗ ІНСТРУМЕНТІВ ДЛЯ СТВОРЕННЯ NLP ЧАТ-БОТА

Селін Я.Ю.

Науковий керівник – ст. викл. каф. ШІ, Гриньова О.Є.

Харківський національний університет радіоелектроніки, каф. ШІ

м. Харків, Україна

e-mail: yaroslav.selin@nure.ua

Examination of different methodologies employed in the development of NLP chatbots is presented in this article. The study compares two primary strategies for chatbot development: constructing from the scratch and utilizing pre-existing no-code platforms. Various criteria, including flexibility, deployment complexity, and customization options, are used to evaluate these approaches. The paper also includes examples of well-known NLP libraries like NLTK and spaCy, as well as off-the-shelf platforms such as Dialogflow and Microsoft Bot Framework. It is emphasized that the selection of an approach depends on factors such as technical expertise, customization and integration requirements, and the specific functional demands of the chatbot.

Використання чат-ботів, що працюють на основі штучного інтелекту, значно зросло в сучасну епоху. Ці комунікаційні агенти призначені для взаємодії з людиною і знайшли застосування в багатьох галузях. Вони служать віртуальними помічниками, представниками служби підтримки клієнтів та виконують інші ролі. Інтеграція обробки природної мови (NLP) стала вирішальною для розвитку чат-ботів, що дозволило їм розуміти наміри користувача за його запитами. Чат-бот отримує значення з вхідних даних, досліджує структуру речень, розпізнає ключові терміни та пов'язує їх із задалегідь визначеними намірами або діями. Така технологія підвищує їхню доступність і зручність для користувачів, що в кінцевому підсумку покращує загальний клієнтський досвід.

Існують різні альтернативи, коли йдеться про розробку чат-ботів. Перша стратегія передбачає проєктування чат-ботів з нуля, як правило, за допомогою мов програмування, таких як Python, включення бібліотек і моделей NLP, таких як NLTK або spaCy. Ці мови програмування та бібліотеки дозволяють розробникам створювати чат-ботів з персоналізованими функціями та адаптованими можливостями NLP. Завдяки прямому кодуванню відповідей і дій чат-бота розробники мають повний контроль над його поведінкою, що сприяє високому рівню адаптивності та кастомізації.

Друга стратегія передбачає використання вже існуючих платформ для розробки чат-ботів, які не потребують складного програмування (no-code).

Natural Language Toolkit (NLTK) та spaCy – дві популярні бібліотеки Python, що використовуються для обробки природної мови (NLP). NLTK, що є стандартною бібліотекою Python, спеціально розроблена для того,

щоб бути зручною для користувача і має на меті спростити реалізацію функцій та утиліт NLP. NLTK пропонує широкий спектр інструментів для різних завдань, таких як токенізація, стеммінг, лематизація, тегування частин мови, розпізнавання іменованих сутностей тощо. NLTK надає розробникам легкий доступ до лінгвістичних даних, дозволяючи маніпулювати ними та проводити різноманітні лінгвістичні аналізи. Вичерпна документація та різноманітні мовні ресурси, які надає NLTK, роблять бібліотеку ідеальним вибором для освітніх цілей, дослідницької діяльності та експериментів у галузі NLP. Багаторічна апробація NLP-спільнотою та відкритий характер дозволяють широко кастомізувати і підтримувати багато мов [1].

Альтернативна бібліотека spaCy є відносно новою, яка фокусується на ефективності та продуктивності. Вона спеціально розроблена для промислового NLP і відома своєю швидкістю та простотою. spaCy оснащена попередньо навченими моделями для різних завдань NLP, таких як тегування частин мови, розпізнавання іменованих сутностей, синтаксичний аналіз залежностей та сегментація речень. Попередньо навчені моделі в бібліотеці дозволяють розробникам швидко створювати та впроваджувати NLP-додатки, не вимагаючи великих навчальних даних. Крім того, spaCy пропонує компоненти зв'язування сутностей і класифікації тексту, що розширює його застосовність до ширшого спектру завдань NLP. Бібліотека оптимізована для промислового використання, що дозволяє швидко розробляти і розгортати NLP-додатки [3].

З іншого боку, готові платформи для розробки чат-ботів, такі як Dialogflow, Microsoft Bot Framework та Amazon Lex, пропонують більш доступне та швидке рішення. Ці платформи пропонують інтуїтивно зрозумілі інтерфейси, готові моделі NLP та ряд інструментів, які дозволяють створювати чат-ботів без глибоких знань програмування. В них використовуються попередньо навчені моделі NLP і пропонується широкий спектр інтеграцій з популярними платформами обміну повідомленнями, що спрощує розгортання чат-ботів на різних каналах.

Dialogflow, що підтримується Google, вирізняється своїми потужними можливостями розуміння природної мови (NLU) за допомогою попередньо навчених моделей машинного навчання. Ця платформа дозволяє розробникам створювати чат-ботів, які демонструють високий рівень розуміння та точності відповідей на запити користувачів. Безперешкодна інтеграція з екосистемою Google та різноманітними платформами обміну повідомленнями підвищує її привабливість, що сприяє широкому розгортанню. Крім того, зручний інтерфейс Dialogflow, розрахований на розробників з різним рівнем навичок машинного навчання, забезпечує доступність для широкого кола користувачів.

Microsoft Bot Framework, як невід'ємний компонент Microsoft Azure, пропонує міцну основу для розробки чат-ботів, оснащену різноманітними

інструментами для управління різними каналами зв'язку, тим самим спрощуючи процес розгортання. Використовуючи когнітивні служби Azure, розробники можуть доповнити своїх ботів розширеними можливостями, такими як розуміння мови, розпізнавання мовлення та навички прийняття рішень. Фреймворк підтримує як кодові, так і графічні підходи до розробки, задовольняючи потреби як технічних, так і нетехнічних спеціалістів [3].

Amazon Lex відомий своїми комплексними послугами штучного інтелекту дозволяє створювати діалогові інтерфейси, які можна легко інтегрувати в різноманітні додатки. Його суттєвою перевагою є безшовна інтеграція з сервісами AWS, що дозволяє розробникам зручно розширювати та контролювати своїх чат-ботів в екосистемі AWS. Можливості розуміння природної мови (NLU) є ключовими для побудови розширених голосових та текстових комунікацій.

Платформи no-code дають змогу користувачам без технічних знань отримати доступ до моделей штучного інтелекту та машинного навчання, сприяючи швидкому створенню прототипів, розробці та розгортанню чат-ботів. Ці платформи надають готові моделі, drag-and-drop інтерфейси та інструменти візуального програмування, які спрощують процес розробки. Як наслідок, вони скорочують час виходу готового продукту на ринок і усувають бар'єри для компаній, які прагнуть впроваджувати рішення на основі ШІ, що є особливо вигідним у сучасному цифровому світі.

Якщо порівнювати ці підходи, то чат-боти, створені на замовлення, забезпечують неперевершену гнучкість і можуть бути налаштовані для конкретних випадків використання, хоча й вимагають значних зусиль при розробці. На противагу цьому, платформи пропонують швидке розгортання, простоту використання, попередньо навчені моделі та інтеграцію з існуючими системами і сервісами, але вони можуть мати обмеження з точки зору кастомізації.

Вибір між створенням чат-ботів з нуля або використанням наявних платформ залежить від таких факторів, як технічний досвід, вимоги до кастомізації, а також конкретні функціональні можливості та інтеграції, необхідні для чат-бота. Обидва підходи використовують концепції NLP, зокрема лінгвістичний аналіз і машинне навчання, щоб дозволити чат-ботам розуміти природну мову, хоча і з різним рівнем ручних зусиль і адаптації.

Список використаних джерел:

1. Natural Language Processing (NLP) in Python with NLTK, Real Python. URL: <https://realpython.com/nltk-nlp-python>.
2. spaCy 101: Everything you need to know, spaCy Usage Documentation. URL: <https://spacy.io/usage/spacy-101>.
3. Microsoft Bot Framework Documentation, Microsoft Bot Framework Developer Portal. URL: <https://dev.botframework.com>.