

Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук _____
Кафедра _____ Медіасистем та технологій _____
Рівень вищої освіти _____ другий (магістерський) _____
Спеціальність _____ 186 Видавництво та поліграфія _____
Тип програми _____ Освітньо-професійна _____
Освітня програма _____ Технології електроних мультимедійних видань _____

(шифр і назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри МСТ _____
(підпис)

«__» _____ 2020 р.

**ЗАВДАННЯ
НА АТЕСТАЦІЙНУ РОБОТУ**

студентові _____ Трофименко Вадима Юрійовича _____
(прізвище, ім'я, по батькові)

1. Тема роботи _____ Розробка методики ефективного використання чат-ботів у WEB _____

Затверджена наказом по університету від 23 жовтня 2020р. № 1432 Ст
1502 – КТСВПВм, 1503 – ТЕМВм

2. Термін подання студентом роботи до екзаменаційної комісії _____ 10 грудня 2020р. _____

3. Вихідні дані до роботи
Національні та міжнародні стандарти оцінки якості веб-сайтів; Методи та принципи побудови інтерфейсу користувача; Вихідні дані до друкарського видання; Методи UML-моделювання

4. Перелік питань, що потрібно опрацювати в роботі
1 Аналітичний огляд досягнень в галузі Web-технологій;
2 Дослідження методів використання чат-ботів у Web
3 Розробка методики ефективного використання чат-ботів у Web;
4 Практична реалізація чат-боту;
5 Експериментальна частина;
6 Економічна частина;
Висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів)
Перелік слайдів презентації: Актуальність та мета роботи; Задачі; Дослідження стану проблеми; Види та класифікація чат-ботів; Система Question-answer; Методика ефективного використання чат-ботів у Web; Практична реалізація чат-боту; Експериментальна частина; Обробка та аналіз результатів експерименту; Економічна частина; Висновки.

6. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	проф. Єгорова І.М.		
Економічна частина	проф. Полозова Т.В.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналітичний огляд досягнень в галузі Web-технологій	27.10.2020	
2	Дослідження методів використання чат-ботів у Web	30.10.2020	
3	Розробка методики ефективного використання чат-ботів у Web	6.11.2020	
4	Практична реалізація чат-боту	15.11.2020	
5	Експериментальна частина	25.11.2020	
6	Економічна частина	28.11.2020	
7	Оформлення пояснювальної записки	30.11.2020	
8	Оформлення графічної частини	10.12.2020	

Дата видачі завдання число місяць 2020 р.

Студент _____ Трофименко В.Ю.
(підпис)

Керівник роботи _____ к.т.н., проф. Єгорова І.М.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка містить 87 сторінок, 16 рисунків, 18 таблиць, 28 використаних літературних джерел.

ЧАТ-БОТ, WEB, МЕТОДИКА, МАТЕМАТИЧНА СТАТИСТИКА, ЕКСПЕРИМЕНТ, ДОСЛІДЖЕННЯ.

Мета роботи: розробка методики ефективного використання чат-ботів у Веб на підставі проведеного дослідження.

Об'єкт дослідження: процес створення та використання чат-боту у веб.

Предмет дослідження: методика ефективного використання чат-ботів у веб.

Для досягнення мети роботи треба виконати наступні задачі:

- провести дослідження видів, структур, принципів роботи та характеристик чат-ботів;
- розробити методику ефективного використання чат-ботів у веб;
- розробити чат-бота з використанням розробленої методики;
- провести експеримент з використанням методів математичної статистики.

Створена методика ефективного використання чат-ботів у веб дозволить значно прискорити процес проектування та розробки чат-ботів.

ABSTRACT

The explanatory note contains: 87 pages, 16 images, 18 tables, 28 literary sources.

CHAT-BOT, WEB, METHODOLOGY, MATHEMATICAL STATISTICS, EXPERIMENT, RESEARCH.

Purpose: to develop a methodology for effective use of chatbots on the Web based on the study.

Object of research: the process of creating and using a chatbot on the web.

Subject of research: methods of effective use of chatbots on the web.

To achieve the goal of the work must perform the following tasks:

- to study the types, structures, principles of operation and characteristics of chatbots;
- to develop a method of effective use of chatbots on the web;
- to develop a chat bot using the developed methodology;
- to conduct an experiment using the methods of mathematical statistics.

A method of effective use of chatbots on the web has been created will significantly speed up the process of designing and developing chatbots.

РЕФЕРАТ

Пояснительная записка содержит 87 страниц, 16 рисунков, 18 таблиц, 28 использованных литературных источников.

ЧАТ-БОТ, WEB, МЕТОДИКА, МАТЕМАТИЧЕСКАЯ СТАТИСТИКА, ЭКСПЕРИМЕНТ, ИССЛЕДОВАНИЕ.

Цель работы: разработка методики эффективного использования чат-ботов в Интернет на основании проведенного исследования.

Объект исследования: процесс создания и использования чат-бота в интернет.

Предмет исследования: методика эффективного использования чат-ботов в интернет.

Для достижения цели работы необходимо выполнить следующие задачи:

- провести исследования видов, структур, принципов работы и характеристик чат-ботов;
- разработать методику эффективного использования чат-ботов в веб;
- разработать чат-бота с использованием разработанной методики;
- провести эксперимент с использованием методов математической статистики.

Созданная методика эффективного использования чат-ботов в веб позволит значительно ускорить процесс проектирования и разработки чат-ботов.

ЗМІСТ

	С.
ВСТУП	10
1 АНАЛІТИЧНИЙ ОГЛЯД ДОСЯГНЕНЬ В ГАЛУЗІ WEB-ТЕХНОЛОГІЙ .	11
1.1 Поняття «Чат-бот» та опис	11
1.2 Історія чат-ботів.....	11
1.3 Тест Т'юрінга	13
1.3.1 Як працює тест Тьюрінга.	13
1.4 Штучний інтелект	15
1.4.1 Ключові характеристики ШІ:.....	16
1.5 Dialogflow	18
1.6 Недоліки чат-ботів	18
1.7 Постановка задачі дослідження.....	20
2 ДОСЛІДЖЕННЯ МЕТОДІВ ВИКОРИСТАННЯ ЧАТ-БОТІВ У WEB	22
2.1 Термін «Чат-бот».....	22
2.2 Види чат-ботів.....	22
2.3 Класифікація чат-ботів.....	23
2.4 Система Question-answer.....	24
2.4.1 Схема роботи систем QA	24
2.4.2 Проблеми систем QA.....	25
2.4.3 Класифікація систем QA	27
2.4.4 Архітектура систем QA	27
2.5 Обробка природньої мови (Natural language processing).....	28
2.6 Інтелектуальний аналіз текстів.....	28
2.6.1 Групи задач ІАТ	28
2.7 Витяг інформації	29
2.7.1 Типові задачі інформаційного витягу.....	31
2.8 Інформаційний пошук.....	31
2.8.1 Задачі Інформаційного пошуку	32

2.9 Типи пошуку	33
2.10 Класифікація текстів.....	35
2.11 Підходи до класифікації текстів	36
2.11.1 Етапи обробки текстів	39
2.12 Інвертований індекс.....	39
2.13 Стоп-слова.....	40
2.14 Стемінг	40
2.15 Лематизація.....	40
2.16 Складання словника	41
2.17 Наївний Байесовський класифікатор.....	42
3 РОЗРОБКА МЕТОДИКИ ЕФЕКТИВНОГО ВИКОРИСТАННЯ ЧАТ-БОТІВ У WEB	46
4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ЧАТ-БОТУ	50
4.1 Підготовчий етап.....	50
4.1.1 Вибір сервісу розробки чат-боту	50
4.1.2 Збір Аналітики сайту.	50
4.1.3 Пошук документації	51
4.1.1 Створення нового акаунту	52
4.2 Розробка.....	52
4.2.1 Створення чат-боту та знайомство з сервісом Dialogflow.....	52
4.2.2 Наповнення текстової бази даних та знайомство зі стандартними intent-ами	54
4.3 Інтеграція та тестування чат-боту	55
5 ЕКСПЕРИМЕНТАЛЬНА ЧАСТИНА	58
5.1 Опрацювання результатів.....	58
5.2 Розрахунки.....	61
5.2.1 Завдання 1	61
5.2.2 Завдання 2	63
5.2.3 Завдання 3	65
5.2.4 Завдання 4	66

5.2.5 Завдання 5	68
5.2.6 Об'єднання результатів тестувань	69
5.2.7 Розрахунок загального математичного очікування та міжгрупової дисперсії	70
6 ЕКОНОМІЧНА ЧАСТИНА	72
6.1 Характеристика науково-дослідної роботи	72
6.2 Етапи виконання НДР, їх трудомісткість та заробітна плата.....	72
6.3 Розрахунок одноразових витрат на розробку НДР	75
6.4 Оцінка результатів науково-дослідної роботи.....	79
6.5 Визначення економічної ефективності результатів НДР.....	80
ВИСНОВКИ.....	82
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	83

ВСТУП

На сьогоднішній день чат-боти знаходяться на передовій штучного інтелекту та людського інтелекту. Сучасні чат-боти можуть зробити все, починаючи від допомоги замовити піцу і закінчуючи складним процесом продажу B2B.

Але різні типи чат-ботів створені для того, щоб робити різні види справ. І це може заплутати. Ще більше може заплутати і сам процес проектування та розробки чат-ботів. Переглянувши велетенські купи документації можна помітити, що ніде не вказано конкретно: як ефективно використати опановані знання для того, щоб розробити реально корисного та діючого чат-бота. Тому аналіз та систематизація інформації про чат-ботів дозволить створити методичку ефективного використання чат-ботів у веб, тим самим роблячи цю роботу актуальною.

Ця методика дозволить значно спростити рівень складності проектування і подальшої розробки чат-ботів, ставши наявним прикладом роботи системи QA. Можна заощадити багато часу під час складання текстової БД питань та відповідей, орієнтуючись на неї.

1 АНАЛІТИЧНИЙ ОГЛЯД ДОСЯГНЕНЬ В ГАЛУЗІ WEB-ТЕХНОЛОГІЙ

В наші дні кількість способів взаємодії з комп'ютеризованими пристроями дуже велика. Один з них – це використання чат-ботів. Вони дуже актуальні на даний момент та навіть у майбутньому.

Так за версією інтернет видання «Theedigital», тренди у ВЕБ на 2021 рік будуть виглядати так [1]:

- більш детальна увага до часу завантаження та швидкості сайту;
- розумне завантаження контенту;
- персоніфікація контенту;
- чат-боти та ін.

Отже можна більш детально розглянути питання чат-боту.

1.1 Поняття «Чат-бот» та опис

Чат-бот – це комп'ютерна програма, написана для участі у бесідах. Зазвичай, чат-боти написані для взаємодії з людьми, та використовуються у дуже великій кількості задач. В бізнесі, вони можуть стати альтернативою веб-сайтам: замість пошуку необхідної інформації на сторінках, користувач може запитати чат-бота, та отримати від нього всю необхідну інформацію. Схоже використовують чат-боти і ті організації, котрі надають велику кількість онлайн інформації для клієнтів, наприклад: організації по захисту здоров'я або державні органи. Для взаємодії з користувачем можуть бути задіяні різні аудіо-візуальні методи спілкування (текст, аудіо та відео)[2].

1.2 Історія чат-ботів

Еліза – це перший чат-бот в історії комп'ютерної науки, котрий був розроблений професором Масачусетського інституту технологій Джозефом

Вейзенбаумом у 1966 році. Вона (Еліза) розуміла слова у користувацькому запиті та відповідала на них заздалегідь підготовленими фразами [3].

Ілюзію розуміння створювало поєднування подібностей. Наприклад, якщо користувач вводив: «Моя мам гарний кухар», система могла відповісти: «Розкажіть більше про вашу родину».

Аліса – була розроблена у 1995 році Річардом Валасом. На відміну від Елізи, чат-бот Аліса могла застосовувати обробку природньої мови, що дозволяло підтримувати більш складні розмови. Це було революційно для відкритого джерела. Розробники могли застосовувати мову розмітки штучного інтелекту (AIML) у своїх власних чат-ботах заснованих на Алісі [4].

Jabberwasky – це розмовник, створений британським програмістом Ролло Карпентером. Його заявлена мета полягає в "імітації природних людських чатів у цікавій, розважальній та жартівливій формі". Це рання спроба створити штучний інтелект за допомогою людської взаємодії.

Заявленою метою проекту є створення штучного інтелекту, здатного пройти тест Тьюрінга. Він призначений для імітації людської взаємодії та проведення розмов з користувачами. Але, Jabberwasky не призначений для виконання будь-яких інших функцій. На відміну від традиційних програм ШІ, технологія навчання призначена як форма розваги, а не для використання в системах комп'ютерної підтримки чи корпоративного представництва.

Останні розробки дозволяють більш сценарійний, контрольований підхід. Основний намір полягає в тому, щоб програма перейшла від текстової системи до повністю голосового управління безпосередньо зі звуку та інших сенсорних входів. Творець Jabberwasky вважає, що його можна вбудовувати в об'єкти будинку, у роботів або в домашніх тварини, що говорять, маючи намір бути корисними та розважальними, складаючи людям компанію.

Міцуку - це чат-бот, створений за технологією AIML Стівом Ворсвіком. Стверджується, що це 18-річна жінка-чат-бот з Лідса, Англія. Він містить усі файли AIML від Аліси, з багатьма доповненнями розмов, створених користувачами, і завжди знаходиться в стадії розробки. Її інтелект включає

здатність думати над конкретними предметами. Наприклад, якщо хтось запитує "Чи можете ви з'їсти будинок?", Міцуку шукає властивості для "будинка". Знаходить значення "made_from" встановлене на "цегла" і відповідає "ні", оскільки цегла не їстівний. Вона може грати в ігри та робити магичні трюки на прохання користувача. У 2015 році було встановлено, що вона проводила бесіди більше ніж двісті п'ятдесят тисяч разів на добу.

1.3 Тест Т'юрінга

Тест Т'юрінга - оманливо простий метод визначення, чи може машина продемонструвати людський інтелект: якщо машина може вступати в розмову з людиною, не будучи виявленою як машина – вона продемонструє людський інтелект [5].

Тест Т'юрінга був запропонований у роботі, опублікованій у 1950 році математиком та піонером обчислювальної техніки Аланом Т'юрінгом. Це стало основним мотиватором теорії та розвитку штучного інтелекту (ШІ).

Ключові ознаки:

- Тест Т'юрінга оцінює навички розмови бота.
- Згідно з тестом, комп'ютерна програма може думати, якщо її відповіді можуть обдурити людину і повірити, що вона також є людиною.
- Не всі приймають достовірність тесту Т'юрінга, але його проходження залишається серйозною проблемою для розробників штучного інтелекту.

1.3.1 Як працює тест Т'юрінга.

Швидкий прогрес у обчислювальній техніці зараз помітний у багатьох аспектах нашого життя. У нас є програми, які миттєво перекладають одну мову на іншу; роботи, які за лічені хвилини прибирають весь будинок; фінансуємо роботів, які створюють персоналізовані пенсійні портфелі, та портативні пристрої, які відстежують рівень нашого здоров'я та фізичної форми.

Все це стало відносно буденним. Зараз на передовій технологій піонером у розвитку є штучний інтелект.

Алан Тьюрінг – британський математик розробив деякі основні концепції інформатики, шукаючи більш ефективний метод розбиття закодованих німецьких повідомлень під час Другої світової війни. Після війни він почав думати про штучний інтелект.

У своїй роботі 1950р. Тьюрінг почав із постановки запитання: "Чи можуть машини мислити?" Потім він запропонував тест, який має допомогти людям відповісти на питання.

Тест проводиться в кімнаті для допитів, якою керує суддя. Піддослідні, людина та комп'ютерна програма, приховані від очей. Суддя проводить бесіду з обома сторонами та намагається визначити, хто з них людина, а «хто» - комп'ютер, виходячи з якості їхньої розмови.

Тьюрінг робить висновок, що якщо суддя не може визначити різницю, комп'ютеру вдалося продемонструвати людський інтелект. Тобто він може думати.

Тест Тьюрінга має своїх недоброзичливців, але він залишається мірою успіху проектів штучного інтелекту.

Оновлена версія тесту Тьюрінга містить більше одного людського судді, який допитує та спілкується в чаті з обома суб'єктами. Проект вважається успішним, якщо понад 30% суддів після п'яти хвилин розмови дійшли висновку, що комп'ютер – це людина.

Премія Лобнера – це щорічне змагання за тестом Тьюрінга, яке було розпочато в 1991 році Х'ю Лобнером, американським винахідником та активістом. Лобнер створив додаткові правила, що вимагають від людини та комп'ютерної програми 25-хвилинних розмов із кожним із чотирьох суддів.

Чат-бот на ім'я Юджин Густман прийнятий деякими як перший, хто пройшов тест Тьюрінга в 2014 році.

1.4 Штучний інтелект

Штучний інтелект (ШІ) відноситься до моделювання людського інтелекту в машинах, які запрограмовані на те, щоб мислити як люди і імітувати їх дії. Цей термін може також застосовуватися до будь-якої машини, яка виявляє риси, пов'язані з людським розумом, такі як навчання та вирішення проблем [6].

Ідеальною характеристикою штучного інтелекту є його здатність раціоналізувати та вживати дії, які мають найбільші шанси досягти конкретної мети.

Коли більшість людей чує термін штучний інтелект, перше, про що вони зазвичай думають, - це роботи. Це тому, що багатобюджетні фільми та романи сплітають історії про машини, які схожі на людину та які руйнують землю. Але ніщо не могло бути так далеко від істини, як ця уява.

Штучний інтелект базується на принципі, згідно з яким людський інтелект можна визначити таким чином, щоб машина могла легко його імітувати та виконувати завдання, починаючи від найпростіших і закінчуючи ще більш складними. Цілі штучного інтелекту включають навчання, міркування та сприйняття.

У міру розвитку технологій попередні показники, що визначали штучний інтелект, застарівали. Наприклад, машини, які обчислюють основні функції або розпізнають текст за допомогою оптимального розпізнавання символів, більше не вважаються втіленням штучного інтелекту, оскільки ця функція тепер сприймається як належне, невід'ємна функція комп'ютера.

ШІ постійно розвивається на користь багатьох різних галузей. Машини проводяться з використанням міждисциплінарного підходу, заснованого на математиці, інформатиці, лінгвістиці, психології тощо.

Алгоритми часто відіграють дуже важливу роль у структурі штучного інтелекту, де в простих додатках використовуються прості алгоритми, тоді як більш складні допомагають сформувати потужний штучний інтелект.

Заявок на штучний інтелект нескінченно багато. Ця технологія може бути застосована до різних галузей. ШІ тестується та використовується в галузі охорони здоров'я для дозування ліків та різного лікування пацієнтів, а також для хірургічних процедур в операційній.

Інші приклади машин зі штучним інтелектом включають комп'ютери, які грають в шахи та самокеровані машини. Кожна з цих машин повинна зважити наслідки будь-якої дії, яку вони вживають, оскільки кожна дія впливатиме на кінцевий результат. У шахах кінцевим результатом є перемога в грі. Для самокерованих автомобілів комп'ютерна система повинна враховувати всі зовнішні дані і обчислювати їх, щоб запобігти зіткненням.

Штучний інтелект також має додатки у фінансовій галузі, де він використовується для виявлення та позначення діяльності в банківській та фінансовій сферах, таких як незвичне використання дебетових карток та великі депозити на рахунках - і все це допомагає відділу анти-шахрайства банку. Програми з ШІ також використовуються, щоб допомогти впорядкувати та полегшити торгівлю. Це робиться завдяки спрощенню оцінки попиту, пропозиції та ціноутворення на цінні папери.

1.4.1 Ключові характеристики ШІ:

- штучний інтелект відноситься до моделювання людського інтелекту в машинах;
- цілі штучного інтелекту включають навчання, міркування та сприйняття;
- ШІ використовується у різних галузях, включаючи фінанси та охорону здоров'я;
- слабкий ШІ, як правило, простий і орієнтований на одне завдання, тоді як сильний ШІ виконує завдання, які є більш складними та схожими на людину.

Штучний інтелект можна розділити на дві різні категорії: слабкий і сильний. Слабкий штучний інтелект уособлює систему, призначену для виконання однієї конкретної роботи. Слабкі системи штучного інтелекту включають відеоігри, такі як шахи, та особистих помічників, таких як Alexa від Amazon та Siri від Apple. «Ви ставите помічнику запитання – він відповідає вам».

Сильні системи штучного інтелекту – це системи, що виконують завдання, які вважаються подібними до людини. Вони запрограмовані на вирішення ситуацій, в які немає можливості втручання людини. Такі типи систем можна знайти в таких областях, як машини з автопілотом, або в операційних лікарень.

З самого початку штучний інтелект потрапляв під пильну увагу науковців та громадськості. Однією із загальних тем є ідея про те, що машини стануть настільки високорозвиненими, що люди не зможуть встигати за їх розвитком, і вони будуть вдосконалювати самі себе, з експоненціальною швидкістю.

У багатьох людей є думка, що машини можуть зламати приватне життя людей і навіть бути озброєними. Інші аргументи обговорюють етику штучного інтелекту і те, чи слід поводитися з такими інтелектуальними системами, як роботи, з тими ж правами, що і з людьми.

Самостійні автомобілі були досить суперечливими, оскільки їх машини, як правило, розроблені для мінімального ризику та найменших втрат. Якщо представити сценарій зіткнення з тією чи іншою людиною одночасно, ці машини розраховали б варіант, який спричинив би найменший збиток.

Ще одне спірне питання, яке виникає у багатьох людей із штучним інтелектом, полягає в тому, як він може вплинути на зайнятість людей. У багатьох галузях, які прагнуть автоматизувати певні робочі місця за допомогою інтелектуальних механізмів, існує занепокоєння, що люди будуть витіснені з робочих місць. Автопілоти можуть усунути потребу в водіях таксі та програмах спільного використання автомобілів, тоді як виробники можуть

легко замінити людську працю машинами, роблячи навички людей більш застарілими.

1.5 Dialogflow

Dialogflow – це платформа для розуміння природної мови, яка використовується для розробки і інтеграції діалогового призначеного для користувача інтерфейсу в мобільні додатки, веб-додатки, пристрої, боти, інтерактивні системи голосового відповіді і пов'язані з ними варіанти використання [7].

Традиційні комп'ютерні інтерфейси вимагають структурованого та передбачуваного введення для нормальної роботи, що ускладнює використання цих інтерфейсів неприродньо, а часом і важко. Якщо кінцеві користувачі не можуть легко зрозуміти цей структурований вхід, їм важко зрозуміти, що робити. В ідеалі інтерфейси можуть зробити висновок про те, чого хочуть кінцеві користувачі, виходячи з природної мови, якою вони користуються [8].

Наприклад, простий запит користувача на зразок: «Який прогноз погоди сьогодні?». Інші кінцеві користувачі також можуть запитати: «Яка зараз погода?», «Яка буде температура завтра в Харкові?», «Яка погода буде 31 грудня?». Навіть маючи ці прості запитання, можна зрозуміти, що спілкування важко реалізувати. Інтерпретація та обробка природної мови вимагає дуже надійного синтаксичного аналізатора. Dialogflow вирішує це, він забезпечує якісний розмовний інтерфейс для кінцевих користувачів.

1.6 Недоліки чат-ботів

Відсутність емоцій.

На відміну від людей, чат-бот не має емоцій. Однак вони дуже важливі, щоб розмова йшла правильно. Керівники служби обслуговування клієнтів

можуть розуміти емоції клієнтів і відповідати відповідним чином, але чат-бот може цього не зробити [9].

Боти можуть бути занадто механічними. Заздалегідь запрограмовані розмови в змозі обробляти запити клієнтів, коли потік розмови йде за певним шляхом. Але вони можуть не впоратися, коли розмова набуває несподіваного повороту.

Оскільки чат-боти не мають почуттів та емоцій, для них може стати критично важливою взаємодія з людьми в деяких випадках.

Дуже розмовні та контекстні чат-боти також можуть запам'ятати певні речі з минулих бесід. Вони можуть скористатися цією статистикою, щоб відповісти відповідним чином і правильно обробляти запити клієнтів.

Складнощі в проектуванні та розробці.

Створити чат-бота з нуля дуже складно. Це вимагає, щоб ви витратили значний час та сили на його створення. Вам також може знадобитися трохи знань з кодування, щоб створити чат-бот, який краще функціонує.

Але, деякі платформи по типу Dialogflow допомагають легко створювати, керувати та вимірювати рішення для обслуговування клієнтів за допомогою штучного інтелекту. Для цього не потрібно мати ніяких знань з кодування.

Розроблені лише для питань першого рівня.

Одним з найбільших недоліків чат-ботів є те, що вони були розроблені для вирішення лише питань першого рівня. Вони можуть не мати змоги вирішувати складні запити. Потрібно навчити їх правильно спілкуватися з клієнтами.

Також потрібно структурувати та оптимізувати базу знань зручним для ботів способом. Це дозволить чат-боту ефективно сканувати вміст та витягувати відповідні рішення для кожного запиту клієнта.

Якщо ресурс не містить відповідей на конкретне запитання будь-якого клієнта, бот може не відповісти на проблеми, які виникають у клієнта.

Постійний контроль та обслуговування

Чат-боти вимагають постійного перегляду, обслуговування та оптимізації з точки зору їхньої бази знань та способу спілкування з клієнтами.

Потрібно наповнювати їх новими, проникливими даними (вмістом), які вони можуть використовувати для відповіді на запити та запитання клієнтів. Вони повинні мати можливість витягувати найбільш релевантний та корисний вміст для вирішення проблем клієнтів.

Для цього потрібно проаналізувати розмови ботів та визначити найпоширеніші запитання, які їм задають клієнти. Потім можна додати відповіді на ці запитання до бази знань.

Але непросто відстежити та проаналізувати кожну розмову чат-бота.

1.7 Постановка задачі дослідження

Характеристикою всього сказаного раніше можна вважати те, що чат-боти – це інструмент, корисність котрого залежить напряду від рівня володіння ним. Невдало спроектований чат-бот може не виправдати очікування від замовника.

Таким чином, в цій роботі можна підкреслити наступні складові:

- мета роботи: розробка методики ефективного використання чат-ботів у Веб на підставі проведеного дослідження;
- предмет дослідження: методика ефективного використання чат-ботів у веб;
- об'єкт дослідження: процес створення та використання чат-боту у веб.

Для досягнення мети роботи треба виконати наступні задачі:

- провести дослідження видів, структур, принципи роботи та характеристик чат-ботів;
- розробити методику ефективного використання чат-ботів у веб;
- розробити чат-бота з використанням розробленої методики;

– провести експеримент з використанням методів математичної статистики.

У наступному розділі науково-дослідної роботи планується більш детально розглянути питання чат-ботів, їх основні характеристики, терміни і т.д.

2 ДОСЛІДЖЕННЯ МЕТОДІВ ВИКОРИСТАННЯ ЧАТ-БОТІВ У WEB

2.1 Термін «Чат-бот»

Чат-бот - це програмний додаток, що використовується для ведення онлайн-чату з допомогою тексту або перетворення тексту в мову, замість забезпечення прямого контакту з живим агентом-людиною. Створені для переконливого моделювання поведінки людини в якості співрозмовника, системи чат-ботів зазвичай вимагають постійного налаштування і тестування, і багато хто з них не може адекватно спілкуватися або пройти стандартний тест Тьюринга.

Термін «ChatterBot» був спочатку введений Майклом Молдін (творцем першого Verbot) в 1994 році для опису цих діалогових програм [10].

Чат-боти використовуються в діалогових системах для різних цілей, включаючи обслуговування клієнтів, маршрутизацію запитів або збір інформації. У той час як деякі додатки чат-ботів використовують великі процеси класифікації слів, процесори природної мови і складний ШІ, інші просто сканують загальні ключові слова і генерують відповіді, використовуючи загальні фрази, отримані з відповідної бібліотеки або бази даних.

Доступ до більшості чат-ботів здійснюється через спливаючі вікна веб-сайтів або через віртуальних помічників. Їх можна розділити на категорії використання, які включають: комерцію (електронна комерція через чат), освіта, розваги, фінанси, здоров'я, новини і продуктивність.

2.2 Види чат-ботів

Виділяють наступні види чат-ботів:

а) за змогою навчатися:

- 1) боти, що навчаються за допомогою машинного навчання;
 - 2) боти, котрих неможна навчити, або скриптові;
- б) за платформою впровадження:
- 1) месенджери;
 - 2) онлайн-консультанти;
 - 3) мобільні додатки;
 - 4) SMS;
 - 5) системи контролю;
 - 6) пошукові системи;
- в) за функціоналом:
- 1) боти-консультанти;
 - 2) боти для збору даних;
 - 3) сервіс-боти.

2.3 Класифікація чат-ботів

Глобально боти ділять на два типи:

- боти, які приймають певні команди;
- боти, котрі здатні аналізувати поступаючу інформацію, та відокремлювати сенс з тексту.

Перший тип ботів, як правило, використовується для вузькоспрямованих цілей і не розрахований на масового споживача. Для працездатності в його алгоритми заздалегідь закладаються певні команди. Проте, до цих пір більшість ботів працюють за даним принципом.

До другої категорії відносяться більш складні програми. Дані боти, як правило, використовують нейронні мережі і глибоке навчання, в зв'язку з чим, здатні приймати будь-які запити користувача, обробляти їх і видавати відповіді. Таким чином, бот здатний спілкуватися на природній для людини мові.

Під нейронними мережами маються на увазі математичні моделі, а також їх програмні або апаратні реалізації, побудовані за принципом 19 організації та функціонування біологічних нейронних мереж - мереж нервових клітин живого організму. Обробка природної мови (Natural Language Processing, NLP) - узагальнене напрямом штучного інтелекту, а також математичної лінгвістики. Тут вивчаються проблеми синтезу природної мови і комп'ютерного аналізу.

2.4 Система Question-answer

Система QA зазвичай використовується в області обробки природної мови. Вона застосована для відповідей на питання в формі природної мови і має широкий спектр додатків. Типові програми включають: інтелектуальну голосову взаємодію, онлайн-обслуговування клієнтів, отримання знань, персоналізований емоційний чат і багато іншого. Більшість питально-відповідних систем можна розділити на: генеративні і пошукові системи відповідей на питання, системи одно- і багатоетапних відповідей на питання, системи відповідей на відкриті питання і системи відповідей на конкретні запитання.

2.4.1 Схеми роботи систем QA

Продуктивність QA системи залежить від ефективності використаних методів аналізу тексту, а також від якості текстової бази – якщо в неї немає відповідей на питання, QA система не зможе знайти відповідь. Чим більша база – тим ліпше, однак, якщо вона вміщує лише корисну інформацію. Великі сховища, такі як інтернет, містять багато надлишкової інформації [12].

Це призводить до наступних моментів:

- через те, що інформація представлена у різних формах, підвищується повнота інформації. QA система з більшою долею інформації знайде відповідь;

- правильна інформація частіше повторюється, тому помилки пошуків можна мінімізувати;

- вдалість пошуку інформації суттєво залежить від достовірності методів аналізу інформації та формування відповідей.

2.4.2 Проблеми систем QA

У 2002 році група дослідників написала план досліджень в області Q&A систем [12].

Передбачалося розглядати наступні питання:

- типи питань;

Різні питання потребують різних методів пошуку відповідей. Тому треба скласти та вдосконалити методичні переліки типів можливих питань.

- обробка питань;

Одну й туж саму інформацію можна запросити різними способами. Потребується створити ефективні методи сприйняття та обробки семантики (сенсу) передбачення. Важливо, щоб програма розпізнавала еквівалентні за змістом питання, незалежно від використовуваних стилю, слів, синтаксичних взаємозв'язків і ідіом. Хотілося б, щоб Q&A-система розділяла складні питання на кілька простих, і правильно трактувала контекстно-залежні фрази, можливо, уточнюючи їх у користувача в процесі діалогу.

- контекстні питання;

Питання задаються в певному контексті. Контекст може уточнити запит, усунути двозначність або стежити за ходом думок користувача по серії питань.

- джерела знань QA-систем;

Перед тим як відповідати на питання, непогано було б довідатися про доступні бази текстів. Які б способи обробки текстів не застосовувалися, ми не знайдемо правильної відповіді, якщо його немає в базах.

– виділення питань;

Правильне виконання цієї процедури залежить від складності питання, його типу, контексту, якості доступних текстів, методу пошуку та ін. - величезного числа факторів. Тому підходити до вивчення методів обробки тексту потрібно з усією обережністю, і ця проблема заслуговує на особливу увагу.

– формування відповіді;

Відповідь має бути якомога більш природним. У деяких випадках достатньо і простого виділення його з тексту. Наприклад, якщо потрібно найменування (ім'я людини, назву приладу, хвороби), величина (грошовий курс, довжина, розмір) або дата («Коли народився Тарас Бульба?») - прямої відповіді досить. Але іноді доводиться мати справу зі складними запитами, і тут потрібні особливі алгоритми злиття відповідей з різних документів.

– відповідь на питання в реальному часі;

Потрібно зробити систему, яка б знаходила відповіді в сховищах за кілька секунд, незалежно від складності і двозначності питання, розміру і обсягом документної бази.

– багатомовні запитання;

Розробка систем для роботи і пошуку на інших мовах (в тому числі автоматичний переклад).

– Інтерактивність;

Найчастіше інформація, пропонована QA-системою в якості відповіді, неповна. Можливо, система неправильно визначила тип питання або неправильно «зрозуміла» його. У цьому випадку користувач може захотіти не лише переформулювати свій запит, але і «порозумітися» з програмою за допомогою діалогу.

– механізм міркувань;

Деякі користувачі хотіли б отримати відповідь, що виходить за рамки доступних текстів. Для цього в QA-систему потрібно додати знання, загальні для більшості областей, а також засоби автоматичного виведення нових знань.

- профілі користувачів QA-систем;

Відомості про користувача, такі як область інтересів, манера його мови і міркування, що мають на увазі за замовчуванням факти, могли б суттєво збільшити продуктивність системи.

2.4.3 Класифікація систем QA

QA системи можна умовно розділити на:

- вузькоспеціалізовані системи контролю, котрі працюють у конкретних областях (наприклад, медицина чи обслуговування автомобілів).

- загальні системи, що працюють з інформацією для всіх знайомих областей, таким чином з'являється можливість вести пошуку в суміжних областях.

2.4.4 Архітектура систем QA

Перші QA системи були розроблені в 1960-х роках і представлені природно-мовними обгортками для експертних систем, орієнтованих на конкретній галузі. Сучасні системи попередньо визначаються для пошуку відповідей на питання у наданих документах із використанням технології обробки природніх мов (NLP)[11].

Сучасні системи QA зазвичай включають особистий модуль - класифікатор питань, який визначає тип питань і, відповідно, очікуваного відповіді. Після цього аналізу система поступово застосовується до викладеного документа все більш складними та тонкими методами NLP, відсилаючи непотрібну інформацію. Самий жорсткий метод – пошук у документах, який передбачає використання системи пошуку інформації для

вибору тексту, що може містити відповідь. Потім фільтр видає фрази, подібні до очікуваної відповіді (наприклад, на питання «Хто...» фільтр поверне шматочки тексту, що містить імена людей). І в самому кінці, модуль вироблення відповідей знайде серед цих фраз правильну відповідь.

2.5 Обробка природньої мови (Natural language processing)

Обробка природньої мови – загальний напрям штучного інтелекту та математичної лінгвістики. Вона вивчає проблеми комп'ютерного аналізу та синтезу природніх мов. Стосовно штучного інтелекту аналіз означає розуміння мови, а синтез — генерацію розумного тексту. Розв'язок цих проблем буде означати створення зручнішої форми взаємодії комп'ютера та людини [13].

2.6 Інтелектуальний аналіз текстів

Інтелектуальний Аналіз Тексту - це область інтелектуального аналізу даних і штучного інтелекту, метою якої є отримання інформації з наборів текстових документів на основі застосування ефективних, з практичної точки зору, методів машинного навчання і обробки природньої мови. В інтелектуальному аналізі тексту використовуються ті ж підходи до обробки даних, що і інтелектуальний аналіз даних, але різниця між цими областями полягає тільки в кінцевих методах і в тому факті, що інтелектуальний аналіз даних має справу з репозиторіями і базами даних, а не з електронними бібліотеками.

2.6.1 Групи задач ІАТ

Ключовими групами завдань ІАТ є: категоризація текстів, отримання інформації та інформаційний пошук, обробка змін в колекціях текстів, а також розробка засобів подання інформації для користувача. [14]

Категоризація документів полягає у віднесенні документів з колекції в одній або декількох груп (класів, кластерам) схожих між собою текстів (наприклад, по темі або стилю). Категоризація може відбуватися за участю людини, так і без нього. У першому випадку, званому класифікації документів, система ІАТ повинна віднести тексти до вже певних (зручних для нього) класів. У термінах машинного навчання для цього необхідно провести навчання з учителем, для чого користувач повинен надати системі ІАТ як безліч класів, так і зразки документів, що належать цим класам.

Другий випадок категоризації називається кластеризацією документів. При цьому система ІАТ повинна сама визначити безліч кластерів, за якими можуть бути розподілені тексти. В машинному навчанні відповідна задача називається навчанням без вчителя. У цьому випадку користувач повинен повідомити системі ІАТ кількість кластерів, на яке йому хотілося б розбити оброблювану колекцію (мається на увазі, що в алгоритм програми вже закладена процедура вибору ознак).

2.7 Витяг інформації

Витяг інформації (Information Extraction) - це задача автоматичного вилучення структурованої інформації з неструктурованих і / або напівструктурованих документів та інших джерел, представлених в електронному вигляді. У більшості випадків ця діяльність стосується обробки текстів на людській мові за допомогою обробки природної мови (НЛП). Недавні дії в обробці мультимедійних документів, такі як автоматичне

додавання анотацій і витяг контенту з зображень / аудіо / відео / документів, можна розглядати як вилучення інформації.

Через складність проблеми сучасні підходи до ІЕ зосереджені на вузько обмежених областях. Прикладом може служити витяг з новинної стрічки звітів про корпоративні злиття, наприклад, позначених формальним ставленням:

$\{\displaystyle \mathrm {MergerBetween} (компанія_{1}, компанія_{2}, дата)\}$ $\{\displaystyle \mathrm {MergerBetween} (компанія_{1}, компанія_{2}, дата)\}$, з пропозиції онлайн-новин, наприклад:

«Вчора компанія Foo Inc. з Нью-Йорка оголосила про придбання Bar Corp.»

Основна мета ІЕ – надати змогу виконувати пошук на раніше неструктурованих даних. Більш конкретна мета - дозволити логічних міркувань робити висновки на основі логічного змісту вхідних даних. Структуровані дані - це семантично чітко визначені дані з обраного цільового домену, інтерпретовані з урахуванням категорії і контексту.

Витяг інформації - це частина великої головоломки, яка пов'язана з проблемою розробки автоматичних методів управління текстом, крім його передачі, зберігання і відображення. Дисципліна інформаційного пошуку (IR – information retrieval) розробила автоматичні методи, зазвичай статистичного характеру, для індексації колекцій документів та класифікації документів. Іншим додатковим підходом є метод обробки природної мови (NLP), який зі значним успіхом вирішив проблему моделювання обробки людської мови, якщо взяти до уваги масштаб завдання. З точки зору складності і акценту ІЕ вирішує завдання між IR і NLP. Що стосується введення, ІЕ передбачає існування набору документів, в якому кожен документ слід шаблоном, тобто описує одну або кілька сутностей або подій способом, аналогічним таким в інших документах, але відрізняється деталями. Як приклад розглянемо групу новинних статей про тероризм в Латинській Америці, кожна з яких імовірно заснована на одному або декількох терористичних актах. Ми також визначаємо для будь-якої даної задачі ІЕ шаблон, який представляє собою (або

набір) фрейм (и) для зберігання інформації, що міститься в одному документі. У прикладі з тероризмом в шаблоні повинні бути слоти, відповідні виконавцю, жертві і зброї терористичного акту, а також дату, коли відбулася подія. Система ІЕ для вирішення цієї проблеми потрібно, щоб «зрозуміти» статтю про атаку досить, щоб знайти дані, відповідні слоти в цьому шаблоні [15].

Сучасне значення ІЕ стосується зростаючого обсягу інформації, доступної в неструктурованій формі. Тім Бернерс-Лі, винахідник всесвітньої мережі, називає існуючий Інтернет мережею документів і виступає за те, щоб більша частина вмісту була доступною як мережа даних. Поки це не сталося, Інтернет здебільшого складається з неструктурованих документів, у яких відсутні семантичні метадані. Знання, що містяться в цих документах, можна зробити більш доступними для машинної обробки за допомогою перетворення в реляційну форму або розмітки за допомогою тегів XML. Інтелектуальний агент, який стежить за стрічкою даних новин, вимагає від ІЕ перетворення неструктурованих даних у щось, з чим можна аргументувати. Типовим застосуванням ІЕ є сканування набору документів, написаних природною мовою, та заповнення бази даних вилученою інформацією [16, 17].

2.7.1 Типові задачі інформаційного витягу

– розпізнавання іменованих елементів (сутностей), наприклад: імен людей, назв організацій, географічних назв, подій, часових і грошових позначень та ін.;

– виділення термінології: знаходження для даного тексту ключових слів і словосполучень (колокацій).

2.8 Інформаційний пошук

Пошук інформації (Informational Retrieval) - це діяльність з отримання ресурсів інформаційної системи, які мають відношення до інформаційної потреби, із сукупності цих ресурсів. Пошуки можуть базуватися на

повнотекстовому або індексуванні на основі вмісту. Пошук інформації - це наука пошуку інформації в документі, пошуку самих документів, а також пошуку метаданих, що описують дані, та баз даних текстів, зображень чи звуків [19].

Автоматизовані системи пошуку інформації використовуються для зменшення того, що називали перевантаженням інформації. IR-система - це програмна система, що забезпечує доступ до книг, журналів та інших документів; зберігає та управляє цими документами. Веб-пошукові системи - це найбільш видимі IR-додатки.

2.8.1 Задачі Інформаційного пошуку

Головна задача ІІ – допомогти користувачам задовільнити його інформаційну потрібність. Так як описати інформаційні потреби користувача технічно непросто, вони формулюються як деякий запит, який представляє з себе набір ключових слів, що характеризує те, що шукає користувач.

Класична задача ІІ, з якої почався розвиток цієї галузі, - це пошук документів, що задовольняють запиту, в рамках деякої статичної колекції документів. Але список завдань ІІ постійно розширюється і тепер включає:

- питання моделювання;
- класифікація документів;
- фільтрація документів;
- кластеризація документів;
- проектування архітектури пошукових систем і призначених для користувача інтерфейсів;
- витяг інформації, зокрема аннотирования і реферування документів;
- мови запитів і ін.

Також, перед двигунами ІІ ставляться деякі завдання по обробці природних мов, що включає в себе морфологічний аналіз, дозвіл лексичної багатозначності і так далі.

2.9 Типи пошуку

Вирізняють наступні типи пошуку:

– повнотекстовий пошук – найпростіший тип пошуку, який дозволяє знаходити документи, що містять зазначені слова, їх різноманітні форми та синоніми, незалежно від того, в якій частині документа вони є. Повнотекстові можливості пошуку також включають використання морфології. Цей режим дозволяє ввести одне слово в рядок пошуку, і текст буде шукатись у всіх його формах слів (наприклад, яблуко – яблука – яблуні – ... тощо). Виходячи з цього, настійно рекомендується встановити морфологічну базу даних для мови, якою буде здійснюватися пошук.

– фразовий пошук – це пошук у реченнях з можливістю обмежити відстань між словами запиту та зафіксувати порядок слів. Він дозволяє використовувати синонімічні ряди для пошуку фраз. Вони допомагають шукати всі можливі відповідні комбінації слів, які відповідають запиту. В випадку, якщо пошук здійснюється за допомогою фрази, що включає слова, для яких вказані синоніми, тоді автоматично сортуються всі можливі варіанти розкладки слів і включаються всі результати, що містять їх, у результати пошуку.

– пошук схожих документів – при "пошуку схожих" в якості запиту використовується весь текст і аналізується ступінь відповідності проіндексованих документів з текстом запиту. Рекомендується використовувати цей тип запитів для ідентифікації однотипних структурованих документів, які включають загальні елементи - звіти, рахунки-фактури, контракти, резюме тощо.

– пошук за атрибутами – використання цього типу пошуку дозволяє шукати документи за їх атрибутами (формат, відправники, одержувачі тощо). Це дозволяє дізнатися в попередженнях про збіг атрибутів перехопленої інформації з атрибутами, зазначеними у попередженні. Наприклад, ви можете

відстежувати активність окремих користувачів домену, IP-адреси, конкретні адреси електронної пошти, документи тощо.

Цей тип пошуку також можна використовувати для відстеження копіювання або передачі файлів у певних форматах. Наприклад, для проектних організацій цей формат - це файли, створені в AutoCad, і ця функція дозволить їм уникнути витоку інформації з підприємства;

– складні запити – можливість створення складних запитів дозволяє гнучко вказати умови, за яких буде виконуватися пошук. Складні запити можуть включати два або більше простих запитів, об'єднаних логічними операторами. Їх слід використовувати в тому випадку, якщо завдання неможливо вирішити за допомогою простих запитів. Складні запити дозволяють поєднувати до 26 простих текстових та атрибутних запитів, використовуючи логічні оператори AND, OR, AND NOT;

– пошук за регулярними виразами – шаблони регулярних виразів дозволяють шукати дані за їх формою, а не за точним значенням. Шаблон дозволяє знаходити всі документи, що містять дані зазначеної структури. Використовуючи складні регулярні вирази (ланцюжки регулярних виразів), ви можете відстежувати передачу інформації з бази даних, що містить безліч полів;

– цифрові відбитки – ця технологія дозволяє виявити наявність будь-якого контрольованого документа в перехопленому трафіку. Технологія пошуку цифрових відбитків пальців раніше була успішно протестована компанією SearchInform у PlagiatInform, продукт, призначений для виявлення плагіату в навчальних роботах, наукових роботах, журналістиці тощо.

Цифрові відбитки дозволяють шукати документи будь-якого розміру та структури, починаючи від різних фінансових звітів і закінчуючи персональними даними працівників, акціонерів або клієнтів компанії. У цьому випадку виявлення документів можливо, навіть якщо вони відрізняються від оригіналу, наприклад, коли документ складається з декількох.

– пошук за словником – цей метод пошуку дозволяє знайти в перехоплених даних документи, що містять слова та фрази із задалегідь сформульованого списку (словника). Ви можете вказати як відсоток слів зі словника, що міститься в документі, так і відсоток документа, що міститься у словнику. Цей спосіб пошуку зручний, перш за все, для виявлення документів, що стосуються специфіки компанії: звітів, бізнес-планів, технологічних розробок.

– пошук за словником синонімів – цей метод пошуку дозволяє знаходити документи в перехоплених даних, що містять слова та фрази із задалегідь сформульованого списку (словника). Ви можете вказати як відсоток слів зі словника в документі, так і відсоток документа у словнику. Цей спосіб пошуку зручний, перш за все, для виявлення документів, пов'язаних зі специфікою компанії: звіти, бізнес-плани, технологічні розробки.

– адресний пошук – процес пошуку документів за суто-формальними ознаками, вказаними у запиті.

Для впровадження адресного пошуку потрібні наступні вимоги:

- 1) присутність точного адресу документа;
- 2) забезпечення жорсткого порядку розміщення документів в сховищі системи та пристрої збереження;

– документальний пошук – процес пошуку в сховищі інформаційно-пошукової системи первинних документів або в базі даних другорядних документів, відповідаючих запиту користувача;

– фактографічний пошук – процес пошуку фактів, відповідаючих інформаційному запиту.

2.10 Класифікація текстів

Комп'ютер не є панацеєю для класифікації. Багато класифікаційних завдань традиційно вирішуються вручну. Книгам у бібліотеці бібліотекар присвоює категорії. Але ручна класифікація дорога при масштабуванні.

Приклад багатоядерних комп'ютерних чіпів ілюструє один альтернативний підхід: найчастіше класифікація за допомогою постійних запитів - які можна сприймати як правила - найчастіше написані від руки.

Наприклад (багатоядерний або багато-ядерний) та (мікросхема або процесор, або мікропроцесор), правила іноді еквівалентні булевим виразам. Правило фіксує певну комбінацію ключових слів, яка вказує на клас. Правила, кодовані вручну, мають хороші масштабні властивості, але їх створення та підтримка з часом вимагає великих зусиль. Технічно кваліфікована особа (наприклад, експерт домену, який добре вмє писати регулярні вирази) може створювати набори правил, які будуть конкурувати або перевищувати точність автоматично згенерованого класифікатора.

2.11 Підходи до класифікації текстів

Існує три підходи до проблеми класифікації текстів [19].

По-перше, класифікація не завжди проводиться за допомогою комп'ютера. Наприклад, у звичайній бібліотеці бібліотекарі книг вручну призначають тематичні заголовки. Така ручна класифікація є дорогою і не застосовується у випадках, коли необхідно класифікувати велику кількість документів на високій швидкості.

Іншим підходом є написання правил, згідно з якими текст може бути віднесений до певної категорії. Наприклад, одне з таких правил може виглядати так: "Якщо текст містить похідні слова та рівняння, класифікуйте його як математику". Фахівець, який знайомий з предметною областю та має навички писати регулярні вирази, може скласти ряд правил, які потім автоматично застосовуються до вхідних документів для їх класифікації. Цей підхід кращий за попередній, оскільки процес класифікації автоматизований і, отже, кількість оброблених документів майже необмежена. Більше того, побудова правил вручну може забезпечити кращу точність класифікації, ніж машинне навчання (див. Нижче). Однак створення та оновлення правил

(наприклад, якщо ім'я поточного президента використовується для класифікації новин, правило потрібно час від часу міняти) вимагає постійних професійних зусиль.

Нарешті, третій підхід заснований на машинному навчанні. У цьому підході набір правил, або, загальніше, критерій прийняття класифікатора тексту, обчислюється автоматично з навчальних даних (іншими словами, класифікатор вивчається). Дані навчання - це низка хороших прикладів документів для кожного класу. У машинному навчанні необхідність розмітки вручну залишається (термін розмітка означає процес присвоєння класу документа). Але розмітка - завдання простіше, ніж написання правил. Крім того, розмітка може виконуватися в звичайному режимі використання системи. Наприклад, у програмі електронної пошти може бути можливо позначити електронні листи як спам, тим самим формуючи навчальний набір для класифікатора, фільтра спаму. Таким чином, класифікація текстів, заснована на машинному навчанні, є прикладом навчання з викладачем, де роль викладача виконує особа, яка задає набір класів і позначає навчальний набір.

При машинному навчанні набір правил, або в більш загальному випадку - критерій прийняття рішення, автоматично виводиться на основі навчальної вибірки. Якщо метод навчання є статистичними, цей підхід називається статистичною класифікацією текстів (Statistical text classification). Для статистичної класифікації текстів потрібно деяка кількість хороших прикладів (або навчальних документів) для кожного класу. Необхідність ручної класифікації також не виключається, оскільки навчальні документи надходять від людини, який їх розмічає. Розмітка (labeling) - це процес вказівки класу кожного документа. Тематична розмітка документів-набагато простіша процедура, ніж складання правил класифікації.

У задачі класифікації текстів задано опис документа $d \in X$, де X - простір документів (document space), і фіксований безліч класів $C = \{c_1, c_2, \dots, c_j\}$

Класи також називаються категоріями (category) та мітками (labels).

Як правило, простір документів X має велику розмірність, а класи визначаються експертами в залежності від програми, як в прикладі China або documents that talk about multicore computer chips. Крім того, задано навчальну множину D (Training set) розмічених документів $\langle d, c \rangle$, де $\langle d, c \rangle \in X \times C$.

Використовуючи метод навчання (learning method), або алгоритм навчання (learning algorithm), ми хочемо отримати класифікатор, або функцію класифікації u (classification function), яка буде показувати документи у класи.

$$u: X \rightarrow C \quad (2.1)$$

Цей метод називається навчанням з учителем (supervised learning), оскільки контролер (Людина, яка визначає класи і готує навчальні документи) грає роль вчителя, керівника процесом навчання. Позначимо метод навчання з учителем буквою "Г" і запишемо "Г (D) = γ ". Метод навчання Г отримує на вхід навчальну множину D і повертає функцію класифікації " γ ". Класифікатор u і метод навчання Г часто не поділяють. Коли говорять "наївний байєсовський класифікатор є стійким", мають на увазі, що наївний байєсовський метод навчання можна застосовувати для вирішення численних завдань навчання, і дуже мало ймовірно, що при цьому буде побудований катастрофічно поганий класифікатор. Однак, коли ми говоримо "помилка наївного байєсівського методу дорівнює 20%", то описуємо експеримент, в якому конкретний класифікатор u (створений за допомогою наївного байєсівського методу навчання) має 20% -ву помилку.

Розглядається тільки завдання однозначної класифікації (one-of problem), в яких документ може бути елементом тільки одного класу.

Мета класифікації текстів - забезпечити високу точність на тестових, або нових, даних.

2.11.1 Етапи обробки текстів

– індексація документів;

Побудова дійсної числової моделі тексту. Наприклад, у багатовимірному віці слів та їх вазі в документах. Змінити розмір моделей.

– побудова та навчання класифікатора;

Є змога використовувати різні методи машинного навчання: дерево прийняття рішень, наївний класифікатор Баєса, штучні нейронні мережі, метод опорних векторів тощо.

– оцінка якості класифікації.

Можна оцінити критерії повноти, точності, порівняти класифікатори за допомогою спеціальних тестових наборів.

2.12 Інвертований індекс

Інвертований індекс – це така структура даних, в якій для кожного терміну в усіх наявних колекціях документів вказані всі колекції документів в яких цей термін зустрічається.

Припустимо що є колекція з 5 текстів:

– t0 = «Найголовнішого очима не побачиш»;

– t1 = «Не все побачиш очима»;

– t2 = «Головного не побачиш тільки очима»;

– t3 = «Складно побачити головне»;

– t4 = «Побачити очима головне».

Для того що б побудувати інвертований список для даної колекції документів, необхідно:

– скласти таблицю послідовності термінів в кожному документі в супроводі відповідних ідентифікаторів документів;

– відсортувати цю таблицю за алфавітом (за зростанням);

- згрупувати однакові терміни по слову і відокремити ідентифікатори документів;
- впорядкувати ідентифікатори документів (в нашому випадку підійде будь-який варіант сортування). [19].

2.13 Стоп-слова

Іноді деякі дуже поширені слова, які не мають значення для інформаційних потреб користувачів, як правило, виключаються з лексикону. Їх називають стоп-словами. Зазвичай для створення списків стоп-слова сортуються за частотою у збірнику (яка дорівнює загальній кількості повторень терміну у збірнику документів), і тоді найчастіше загальні терміни часто фільтруються вручну на основі їх семантичних посилок на тематична область індексованих документів включена до списку слів зупинок (stop-list), елементи яких відкидаються під час індексації [19].

2.14 Стемінг

Стемінг – наблизений евристичний процес, в ході якого від слів відкидаються закінчення в розрахунку на те, що в більшості випадків це себе виправдає. стемінг часто має на увазі видалення похідних афіксів [19].

2.15 Лематизація

Лематизація – це точний процес з використанням лексикону і морфологічного аналізу слів, в результаті якого видаляються тільки флексивні закінчення і повертається основна, або словникова, форма слова, яка називається лемою [19].

2.16 Складання словника

Враховуючи інвертований індекс та запит, першим завданням є визначити, чи кожен термін запиту існує у словниковому запасі, і якщо так, визначити вказівник на відповідні повідомлення [19].

Дерева пошуку долають багато з цих проблем - наприклад, вони дозволяють нам перерахувати всі словникові терміни, що починаються з автоматичного. Найвідомішим деревом пошуку є бінарне дерево, в якому кожен внутрішній вузол має двох дітей (рис. 2.1). Пошук терміну починається з кореня дерева. Кожен внутрішній вузол (включаючи корінь) представляє двійковий тест, на основі результату якого пошук переходить до одного з двох піддерев нижче цього вузла. наводить приклад бінарного дерева пошуку, що використовується для словника. Ефективний пошук (із кількістю порівнянь, що дорівнює $O(\log M)$), шарніри на дереві, що врівноважується: кількість термінів у двох піддеревах будь-якого вузла або рівна, або відрізняється на одиницю. Основне питання тут полягає в перебалансуванні: оскільки терміни вставляються або видаляються з бінарного дерева пошуку, його потрібно збалансувати, щоб зберегти властивість балансу. Щоб пом'якшити перебалансування, один із підходів полягає в тому, щоб дозволити кількості піддерев під внутрішнім вузлом змінюватися у фіксованому інтервалі. Деревом пошуку, яке зазвичай використовується для словника B-TREE, є B-дерево - дерево пошуку, в якому кожен внутрішній вузол має певну кількість дочірніх елементів в інтервалі $[a, b]$, де a та b є відповідними додатними цілими числами; На малюнку 2.2 показаний приклад з $a = 2$ та $b = 4$. Кожна гілка під внутрішнім вузлом знову представляє тест для діапазону послідовностей символів, як у прикладі двійкового дерева на рисунку 2.1. B-дерево може розглядатися як "згортання" декількох рівнів бінарного дерева в один; це особливо вигідно, коли частина словника знаходиться на диску, і в цьому випадку це згортання виконує функцію попереднього отримання неминучих двійкових тестів. У таких випадках цілі числа a і b визначаються розмірами

дискових блоків. Слід зазначити, що на відміну від хешування, дерева пошуку вимагають, щоб символи, що використовуються у колекції документів, мали встановлений порядок; наприклад, 26 букв англійського алфавіту завжди перераховані в певному порядку від А до Z. Деякі азіатські мови, такі як китайська, не завжди мають унікальне впорядкування, хоча на сьогодні всі мови (включаючи китайську та японську) прийняли стандарт система впорядкування для їх наборів символів.

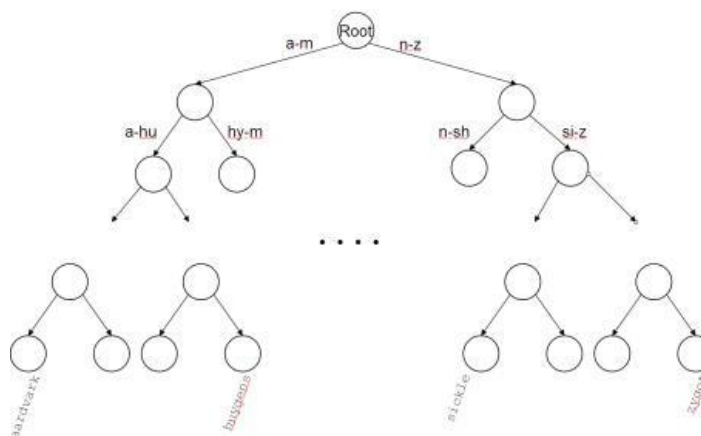


Рисунок 2.1 – Бінарне дерево.

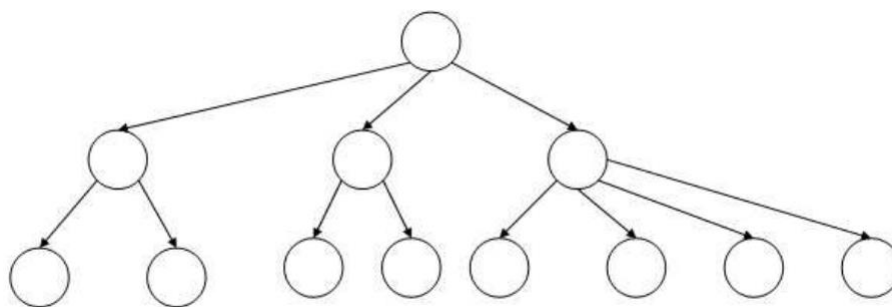


Рисунок 2.2 – В-дерево (дерево пошуку).

2.17 Наївний Байєсовський класифікатор

Метод навчання з учителем, який розглядається – поліноміальний наївний метод Байєса (Naive Bayes - NB).

У цьому методі ймовірність того, що документ d належить класу c , обчислюється таким чином:

$$P(c|d) \propto P(c) \prod_{1 \leq k \leq n_d} P(t_k|c), \quad (2.2)$$

де $P(t_k|C)$ - умовна ймовірність того, що термін t_k з'явиться в документі з класу C .

$P(t_k|C)$ - це оцінка вкладу терміну t_k в те, що документ належить класу C , а $P(C)$ - апіорна ймовірність того, що документ належить класу C . Якщо терміни документа не дозволяють чітко відокремити один клас від іншого, то слід вибрати той з них, який має вищу апіорну ймовірність. Послідовність 2, - складається з лексем документа d , що є частиною лексикону, використовуваного для класифікації, а n_d - кількість таких лексем в документі d . Наприклад: послідовність для документа Beijing and Taipei join the WTO, складається з одного речення, та має вигляд <Beijing, Taipei, join, WTO>, де $n_d = 4$, якщо видалити стоп-слова «and» та «the».

Мета класифікації текстів - знайти найкращий клас для документа. У методі NB найкращим вважається найбільш ймовірний клас, або клас старий, що має максимальну апостеріорну ймовірність (maximum a posteriori - MAP). Можна лише оцінити їх за допомогою навчальних множин.

$$C_{max} = \arg \max_{c \in C} \hat{P}(c|d) = \arg \max_{c \in C} \hat{P}(c|d) \prod_{1 \leq k \leq n_d} \hat{P}(t_k|c), \quad (2.3)$$

У рівності (2.3) перемножується кілька умовних ймовірностей, по одній для кожного значення $1 \leq k \leq n_d$. Це може привести до втрати значущих розрядів. Отже, краще замінити множення їх ймовірностей складанням їх логарифмів. Клас з найбільшим значенням логарифма ймовірності залишається найбільш ймовірним, так як $\log(xy) = \log(x) + \log(y)$, і логарифмічна функція монотонна. Отже, в наївному методі Байєса насправді потрібно знайти точку максимуму наступної функції.

$$C_{max} = \arg \max_{c \in C} [\log \hat{P}(c) + \sum_{1 \leq k \leq n_d} \hat{P}(t_k | c)], \quad (2.4)$$

Рівність (2.4) допускає просту інтерпретацію. Кожен логарифм умовної ймовірності $\log \hat{P}(t_k | c)$ – це вага, яка вказує наскільки важливий термін t_k для класу C . Аналогічно завжди апіорна ймовірність $\log \hat{P}(c)$ це вага, що характеризує відносну частоту класу C . Часто зустрічаються класи, швидше за все є більш правильними, ніж ті, котрі зустрічаються рідше. Таким чином, сума логарифмів апіорної ймовірності і ваг термінів відображає свідчення на користь того, що документ належить класу, а рівність (2.4) ідентифікує клас, якому відповідає би найбільша кількість свідчень.

Для оцінки ймовірностей $\hat{P}(c)$ і C , перш за все треба отримати оцінку максимальної правдоподібності, яка представляє собою відносну частоту і відповідає найбільш імовірною величиною кожного параметра при заданих навчальних даних. Для апіорних ймовірностей оцінка має такий вигляд:

$$\hat{P}(c) = \frac{N_c}{N} \quad (2.5)$$

Де N_c – кількість документів у класі C , та N – загальна кількість документів.

Оціним умовну ймовірність $\hat{P}(t|c)$ як відносну частоту терміну в документах, що належать класу c .

Тут T_{ct} - кількість появ терміна t у обдати документах з класів C з урахуванням багаторазових появ терміна в документі. Ця оцінка заснована на припущенні про позиційну незалежність (positional independence assumption). T_{ct} - це кількість появ терміна у всіх до координатах в документах з навчальної множини. Таким чином, ми не обчислюємо різні оцінки для різних координат, і, наприклад, якщо слово двічі зустрічається $K_1 K_2$, то $\hat{P}(t_k | c) = \hat{P}(t_k | c)$.

З оцінкою максимальної правдоподібності пов'язана одна проблема: якщо пара "термін-клас" не зустрічається в навчальних даних, то оцінка MLE дорівнює нулю. Наприклад, якщо термін WTO в навчальних даних зустрічається тільки в документах класу China, то оцінки MLE для інших класів, наприклад класу UK, дорівнюють нулю. $\hat{P}(WTO|UK) = 0$

Тепер умовна ймовірність приналежності документа Britain is a member of the WTO, що складається з одного речення, класу UK буде дорівнює нулю, оскільки в рівність (2.2) ми перемножуємо умовні ймовірності для всіх термінів. Очевидно, що модель повинна привласнювати класу UK високу ймовірність, оскільки в реченні зустрічається термін Britain. Проте не можна просто відкинути нульову ймовірність для терміну WTO незалежно від того, наскільки багато є свідчень на користь класу UK, забезпечених іншими ознаками. Ця оцінка дорівнює нулю через рідкість терміна. Навчальні дані ніколи не бувають великими настільки, щоб частота рідкісних термінів оцінювалася адекватно, як, наприклад, частота терміна WTO в документах класу UK. Для того щоб позбутися від нуля, ми використовуємо згладжування Лапласа (Laplace smoothing), просто додаючи одиницю до кожної частоти.

$$\hat{P}(t|c) = \frac{T_{ct}+1}{\sum_{ct}(T_{ct}+1)} = \frac{T_{ct}+1}{\sum_{ct} T_{ct}+B} \quad (2.6)$$

Тут $B = |V|$ – кількість термінів в лексиконі колекції. Згладжування Лапласа можна інтерпретувати як апіорне рівномірний розподіл (кожен термін зустрічається в кожному класі по одному разу), яке потім уточнюється на основі надходять навчальних даних. Треба відзначити, що це - завжди апіорна ймовірність появи терміну, а не класу, яка оцінюється формулою (2.6) на рівні документу.

3 РОЗРОБКА МЕТОДИКИ ЕФЕКТИВНОГО ВИКОРИСТАННЯ ЧАТ-БОТІВ У WEB

Аналізуючи інформацію, стосовно принципів роботи чат-ботів, котрі навчаються з вчителем, можна підкреслити наступні шаги, котрі виконуються кожен раз:

- користувач формує питання;

Воно може бути складним, або простим.

Якщо це просте питання, тоді система починає відбір фраз, з текстової бази даних, котра потенційно може містити відповідь.

Після відбору, настає етап фільтрації фраз для відповіді.

Після фільтру генерується відповідь, котра повинна задовільнити користувача.

Складні питання, спочатку розбиваються на прості, або на навідні.

Прості питання відправляються до текстової бази даних, де проводиться вибір та фільтрація фраз, після чого генерується відповідь.

Сформовані навідні питання знов відправляються користувачу проходячи блок перевірки кількості навідних питань, для більш детальної конкретизації. Цей блок необхідний, щоб кількість таких питань не була нескінченною. Зазвичай їх не більше 4-х. Якщо більше, тоді чат-бот відповідає по типу: «пробачте, я Вас не зрозумів».

Завдяки цьому досягається формулювання простого питання, на яке буде згенерована відповідь.

Таким чином, можна узагальнити поведінку системи на:

- прийняття запиту від користувача;

- відповідь одразу, якщо це просте запитання, або:

- приведення до простого, якщо це складний запит, використовуючи розбиття складних запитання на прості, або наведення нових запитань для уточнення;

– відповідь.

Для більшої наявності можна скласти діаграму принципу роботи чат-бота, котра відображена на рисунку 3.1.

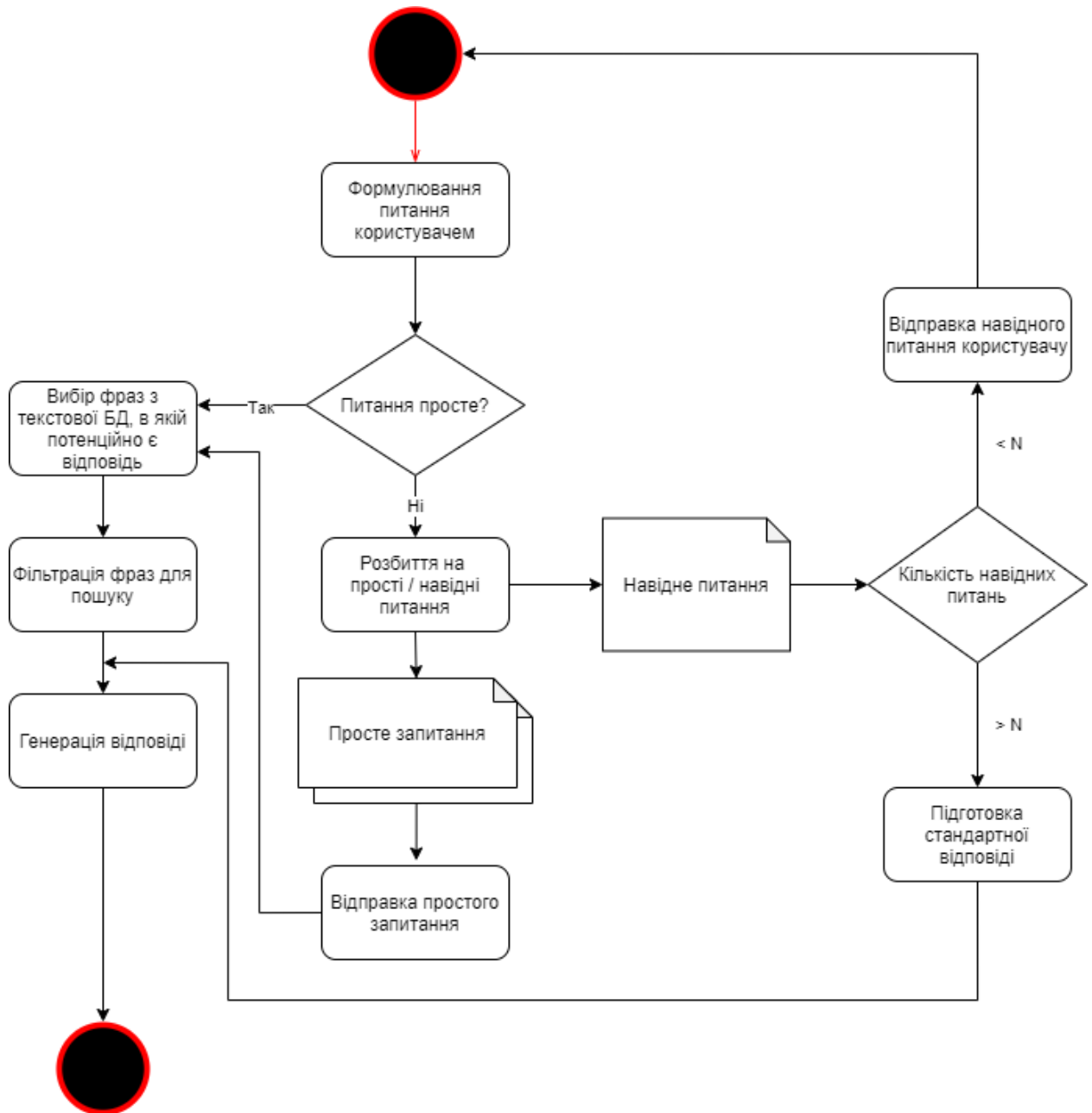


Рисунок 3.1 – Принцип роботи чат-бота, який навчається з вчителем

Розглянувши принцип роботи чат-бота, можна віднести його до простої QA-системи, яка складається з наступних модулів:

- класифікатор питань;
- модуль пошуку в документах;

- фільтр;
- модуль виділення відповідей.

Для більшої наявності, було створено діаграму компонентів QA-система, яка зображена на (рис. 3.2)



Рисунок 3.2 – Компоненти QA-системи

Аналізуючи попередню інформацію, доцільно говорити, що ефективність чат-бота залежить від правильності складання запитально-відповідальної бази. Для цього логічно провести аналіз поведінки користувачів на сайті. Тобто, які питання вони найактивніше задають, або які сторінки частіше переглядають.

Для отримання інформації поведінки користувача використовуються спеціальні сервіси збору статистики, наприклад, Google Analytics.

Отримана інформація підлягає обов'язковому аналізу. Цей процес відображає картину поведінки користувача на сайті, на підставі котрої формується БД питання-відповідь.

Після формування БД, слід провести інтеграції чат-бота на веб-сайт. Але це не останній етап. Як було сказано в попередніх розділах, чат-бот – це система, котра потребує постійної підтримки та вдосконалення. Тому після впровадження слід постійно доповнювати БД для більш корельованих відповідей.

Вище описану методику можна відобразити як діаграму, котра зображена на (рис. 3.3).

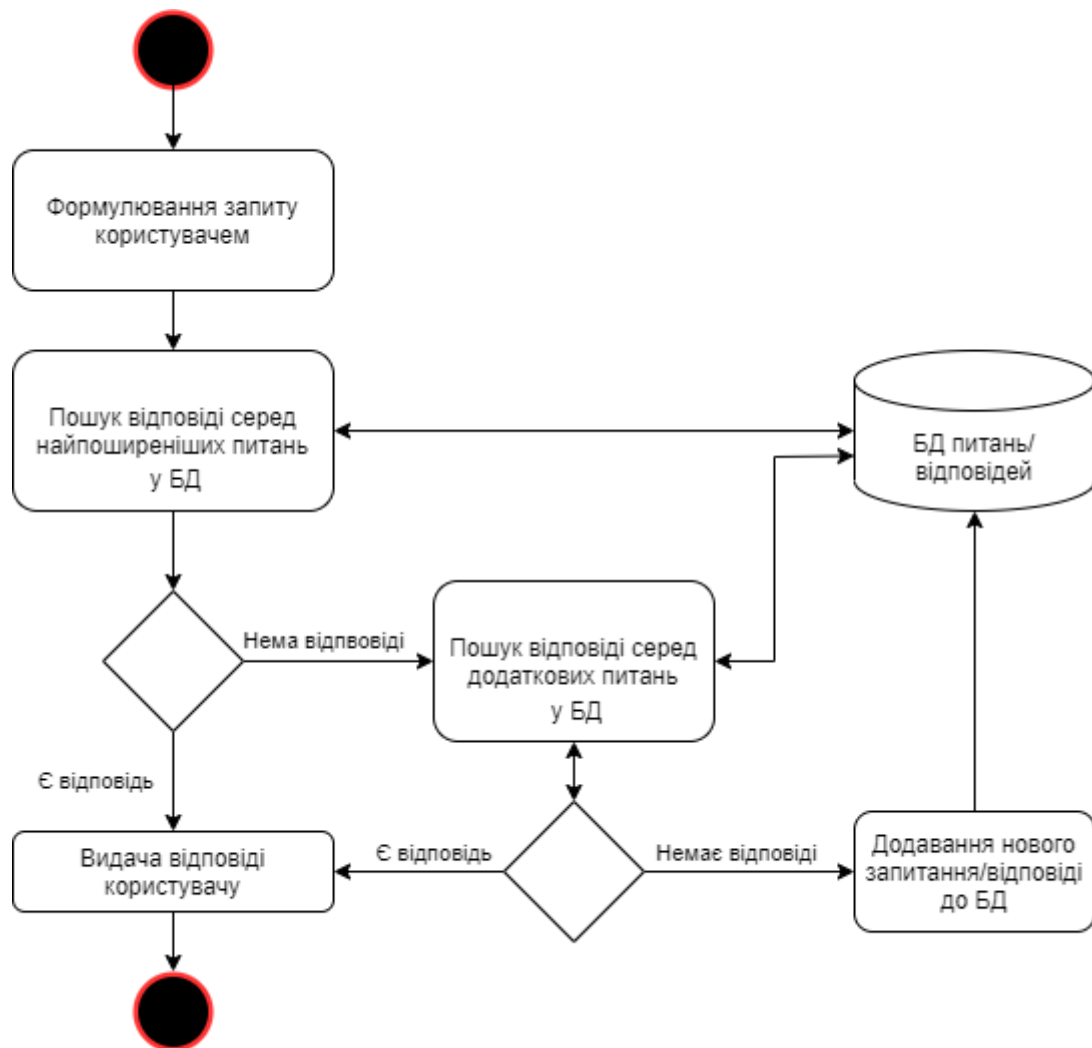


Рисунок 3.3 – UML-діаграма ефективного використання чат-боту у веб

4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ЧАТ-БОТУ

4.1 Підготовчий етап

4.1.1 Вибір сервісу розробки чат-боту

Для реалізації чат-боту було обрано сервіс Dialogflow. Як було сказано у попередніх розділах, цей сервіс надає велику кількість можливостей для створення чат-ботів з легкою інтеграцією у месенджери та на сайти.

4.1.2 Збір Аналітики сайту.

Користуючись сервісами для збору статистики по типу Google Analytics можна відстежити частоту ключових слів, карту відвідувань і т.д., що буде відображати інтереси користувачів. Умовно кажучи статистика відвідувачів допоможе в складанні текстової БД. Чим більше приділяється увага до найчастіших питань, тим точніше буде відповідати чат-бот [20].

Дані з сайту Google Analytics приведені на (рис. 4.1– 4.2).

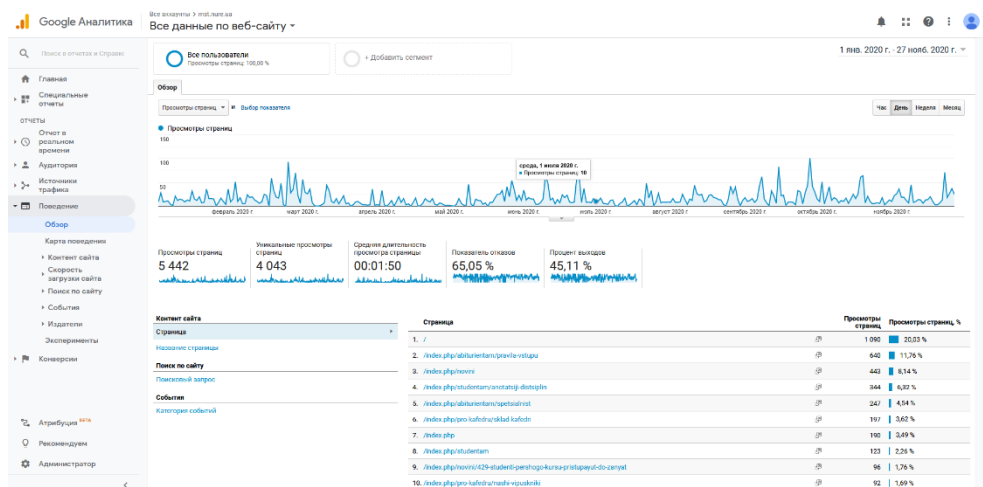


Рисунок 4.1 – Найбільш відвідувані сторінки

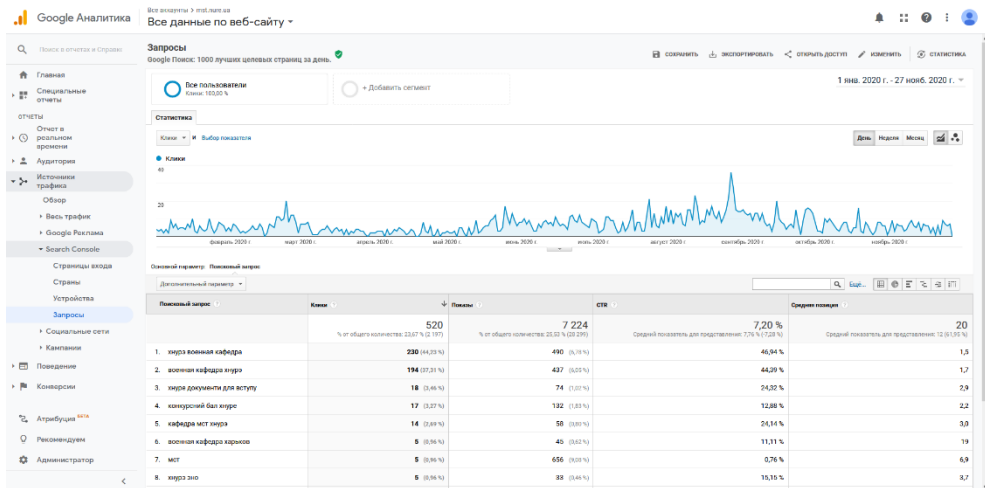


Рисунок 4.2 – Топ ключових слів

4.1.3 Пошук документації

Перед початком, як правило дуже доречним є не тільки обрання засобів для розробки, а й пошук документації, що значно полегшить знаходження відповідей на питання, котрі з'являються в процесі.

Корпорація Google піклується про свої сервіси, тому присвятила цілий сайт з інформативними документами для них.

Приклад роботи з інформативними документами приведений на (рис. 4.3)

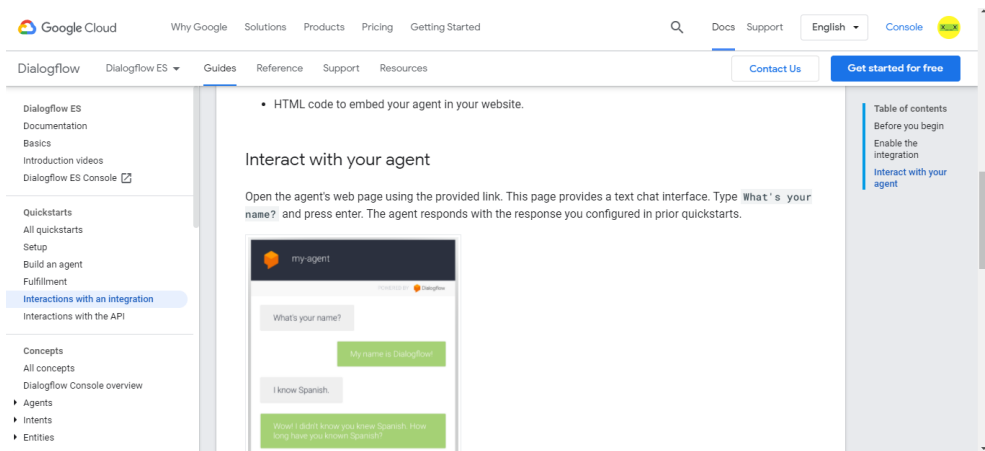


Рисунок 4.3 – Приклад документації до Dialogflow

4.1.1 Створення нового акаунту

Гарним тоном у веб-розробці «Під ключ» є створення нового акаунту для проекту. Отже з цього і починається підготовчий етап.

Приклад реєстрації нового акаунту приведений на (рис. 4.4).

Рисунок 4.4 – Реєстрація нового Google акаунту

4.2 Розробка

4.2.1 Створення чат-боту та знайомство з сервісом Dialogflow

Після авторизації, сервіс одразу пропонує ознайомитися з документацією – що дуже зручно. Але в даний випадок треба натиснути на кнопку «CREATE AGENT». Як на (рис. 4.5).

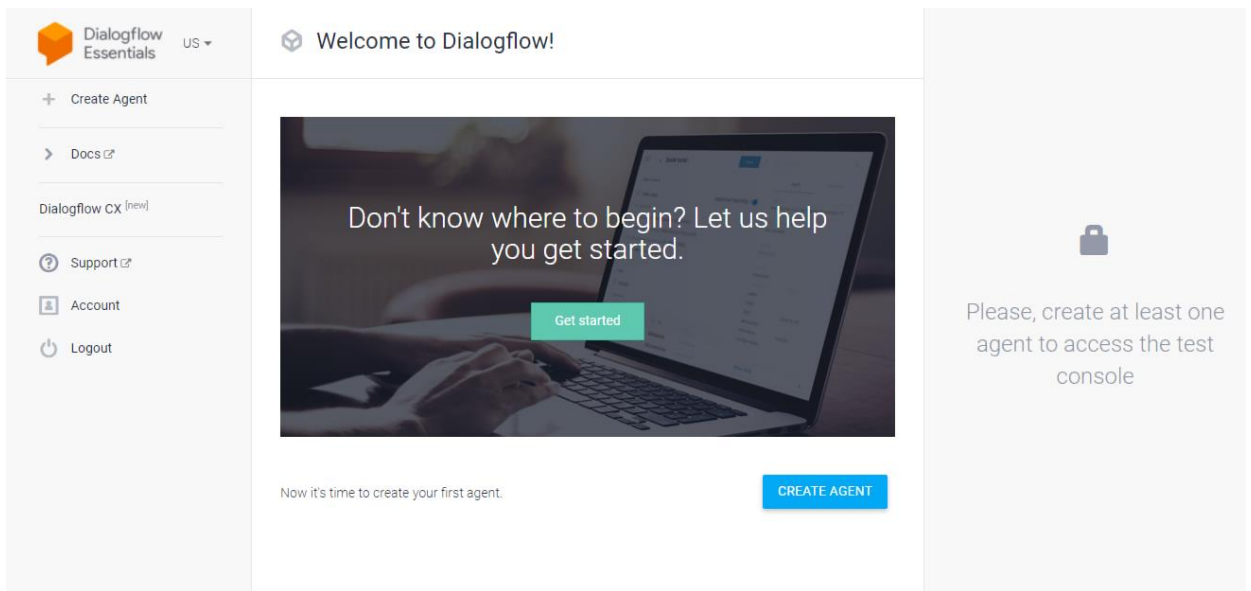


Рисунок 4.5 – Вікно привітання сервісу Dialogflow

Далі слід дати розроблюваному чат-боту назву, та обрати мову як на (рис. 4.6).

Dialogflow постійно навчається і розпізнає текст з урахуванням особливостей, тому правильне вказування мови підвищить точність розпізнавання.

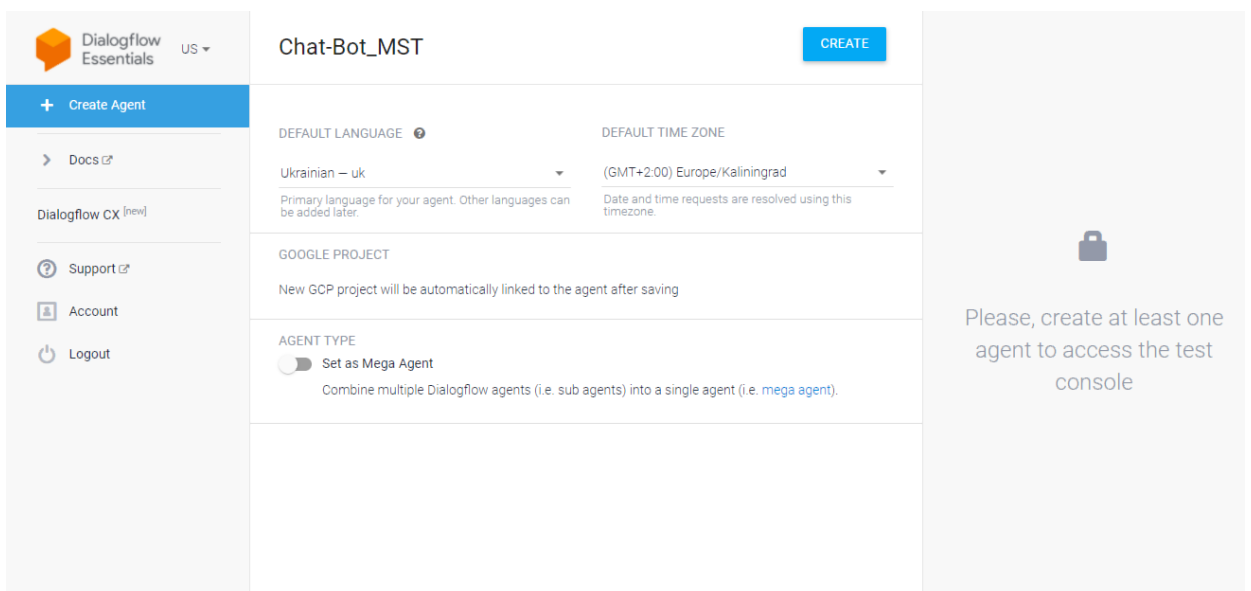


Рисунок 4.6 – Назва та вибір мови

4.2.2 Наповнення текстової бази даних та знайомство зі стандартними intent-ами

З самого початку нам доступні 2 наміри (intent):

- default Welcome intent;
- default Fallback intent.

Default Welcome intent – намір, котрий вітає користувачів незалежно від платформи, на якій використовується чат-бот. Для більшої наявності на (рис.4.7 – 4.8), показано приклад процесу редагування наміру привітання відвідувачів сайту.

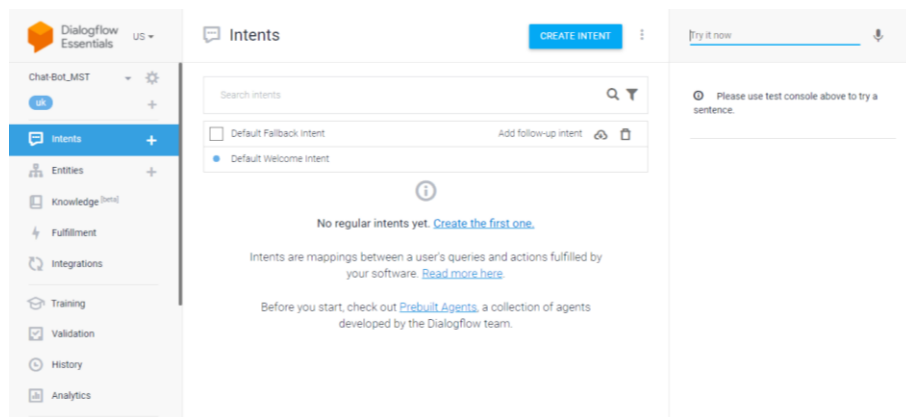


Рисунок 4.7 – Клік на Default Wellcome intent

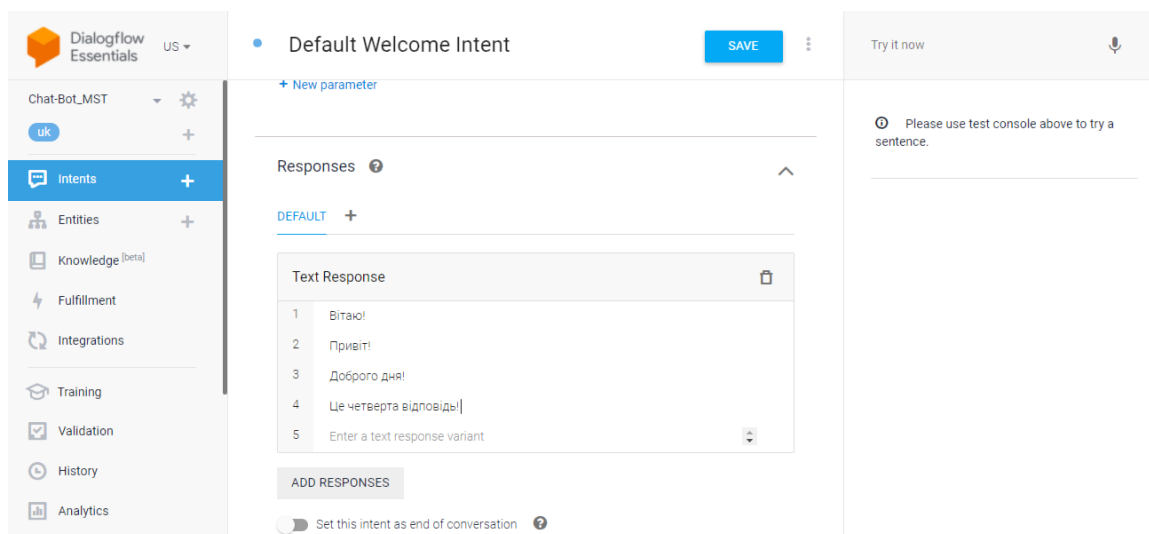


Рисунок 4.8 – Редагування фраз-вітань

На дному етапі вже можна говорити, що чат-бот створено, але він ще не наповнений, тому слід зробити це. Створюючи нові наміри – розширюється текстова БД питань-відповідей. Чим вона більша і логічніша, тим точніші для кінцевого користувача відповіді на його питання.

4.3 Інтеграція та тестування чат-боту

Після розробки чат-боту, його слід протестувати, щоб виявити початкові недоліки, баги та ін.

Для того щоб в повні мірі протестувати бота, його слід інтегрувати на сайт. На щастя, сервіс DialogFlow дозволяє зробити це майже в один клік. Перейшовши на вкладинку «Integrations», Обираємо «Dialogflow Messenger», у відкритому вікні можна побачити код для інтеграції його на сайт, як на (рис. 4.9).

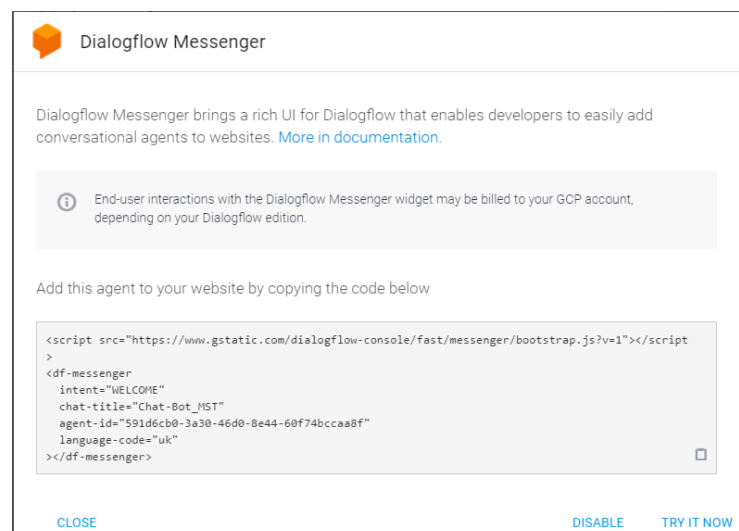


Рисунок 4.9 – Код для інтеграції чат-боту на сайт

В процесі тестування слід переконатися, що все працює як розраховувалося, якщо ні, то слід повернутися на попередні етапи, та скорегувати недоліки.

Процес тестування приведений на (рис. 4.10 – 4.11).



Рисунок 4.10 – Тестування привітання користувача

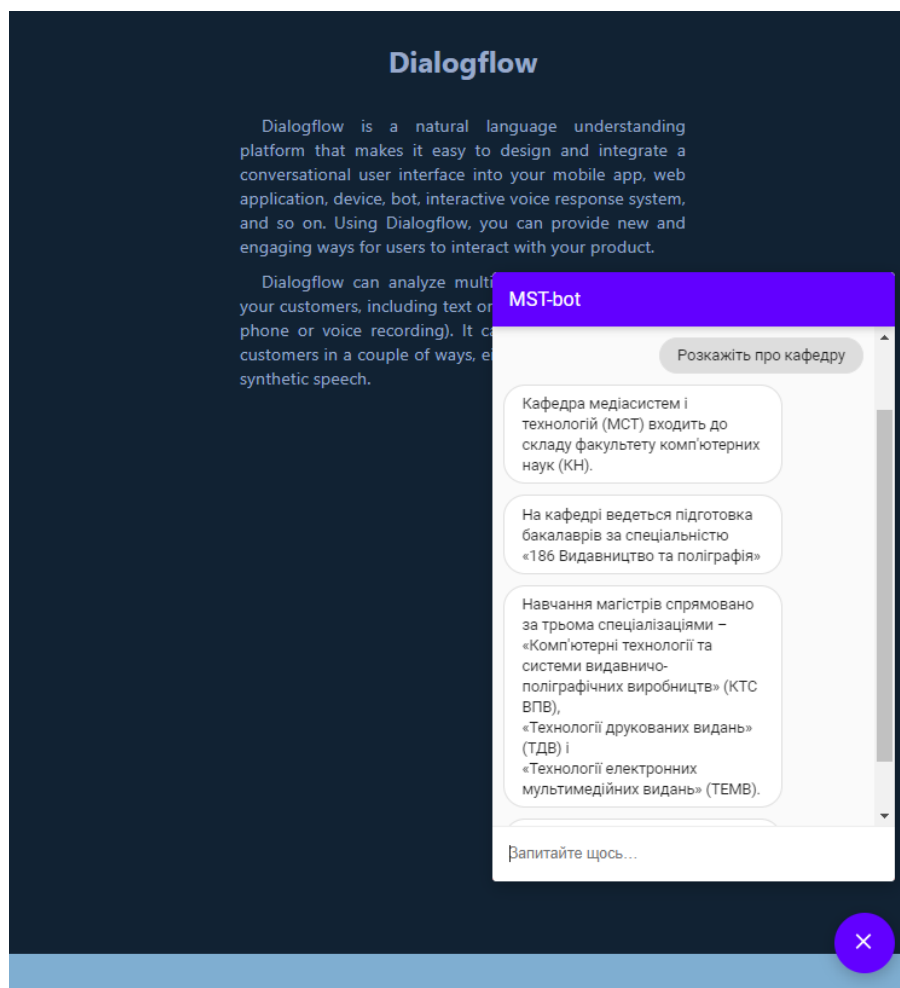


Рисунок 4.11 – Запит у чат-бота «Розкажіть про кафедру»

Таким чином, в даному розділі було спроектовано та розроблено чат-бот за допомогою сервісу Dialogflow. Головною метою розробки було чітке слідування розробленої методики ефективного використання чат-ботів у ВЕБ.

5 ЕКСПЕРИМЕНТАЛЬНА ЧАСТИНА

Для проведення тестування було відібрано 20 учасників, різного віку та професій. Кожен з них ніколи не бачив сайт кафедри МСТ. Їм було запропоновано виконати пошук відповідей на п'ять заздалегідь підготованих питань.

Пошук треба здійснити на сайті кафедри МСТ двома способами:

Перший спосіб: полягає в пошуку відповідей вручну (пошук здійснюється лише переглядом усіх сторінок, на яких можливо є відповідь).

Другий спосіб: використання чат-боту для пошуку інформації.

Мета тестування полягала в тому, щоб зробити виміри часу, витраченого на виконання кожного завдання, тим самим отримати данні про ефективність впровадження чат-боту на сайт.

5.1 Опрацювання результатів

Вимірювання часу починалося одразу після оголошення питання особі, котра тестується. В експерименті брали участь 20 чоловік, що відповідає методу малих вибірок.

Перше тестування А виконувалося лише вручну та не передбачало використання ніяких методів прискорення пошуку інформації на сайті.

Під час другого тестування Б, застосовувався чат-бот, розроблений за допомогою сервісу DialogFlow. Кожен тестувальник повинен був знайти по п'ять відповідей під час проходження тесту А та тесту Б, відповідно.

Час виконання завдання – кількість часу, котра використана на пошук відповіді з моменту оголошення питання. Час виконання було зафіксовано вручну за допомогою секундоміру. Вся зібрана інформація від експерименту була зведена до табличного вигляду. На підставі цих даних були розраховані математична дисперсія та математичне очікування.

Результати виконання експерименту наведені нижче в (табл. 5.1– 5.2).

Таблиця 5.1 – Результати першого тестування А (без чат-бота).

№ учасника	Час виконання завдання (сек)				
	1	2	3	4	5
1	22.9	50.6	57.2	36.3	64.9
2	44.2	59.9	36.0	49.9	46.5
3	48.8	34.8	56.9	70.1	42.2
4	42.5	24.7	57.4	31.0	61.9
5	29.4	31.4	44.4	21.4	66.4
6	54.8	59.1	53.0	46.7	42.8
7	35.6	41.3	43.8	42.8	46.9
8	52.0	60.3	21.3	54.2	35.7
9	31.3	34.3	43.1	49.1	62.6
10	28.2	53.5	60.7	41.0	49.5
11	28.1	36.4	50.1	70.1	27.5
12	47.6	37.1	68.0	36.4	68.6
13	62.9	59.8	34.6	44.7	61.0
14	22.3	53.6	55.0	21.0	63.9
15	34.9	41.1	22.4	28.9	45.1
16	66.2	41.0	51.9	27.2	48.8
17	28.2	51.0	62.5	65.5	63.2
18	47.1	59.8	43.1	56.6	29.8
19	27.0	60.7	42.2	64.4	70.5
20	59.9	52.7	21.8	61.8	21.6

Таблиця 5.2 – Результати другого тестування Б (з чат-ботом).

№ учасника	Час виконання завдання (сек)				
	1	2	3	4	5
1	16	27.3	10.1	6	11
2	8.1	5.7	6.2	4.2	6.4
3	15.2	19.2	3.9	10.7	12.2
4	17.1	14.1	7.8	5.2	5.3
5	20.2	4.3	11.4	9.8	8.4
6	12.5	7.6	8	3.6	15.3
7	16.1	15.3	6.3	2.8	10.2
8	12.9	17.5	12.6	9	14.5
9	19.7	13.4	3.2	6.1	8.6
10	11.8	21.1	5.1	13.5	11.7
11	19.4	18.2	15.8	2.5	5.5
12	15.6	15	6.4	7.4	9.1
13	16.3	12.3	2.4	11.3	7
14	18.5	23.1	6.2	6.1	3.9
15	7.9	15.5	3.8	13.6	7.7
16	9.2	17.3	10.4	8	15.8
17	5.8	12.2	9.2	2.8	12.4
18	11.3	14.6	5.6	5.7	9.7
19	6.3	16.8	10.8	11.1	5.2
20	7.7	3	5.3	10.7	17.8

Для коректного опрацювання даних треба узгодити час. Так як час – це випадкова величина, котра не може бути прогнозована заздалегідь, треба розрахувати математичне очікування неперервної випадкової величини, формула якої:

$$M[X] = \sum_{i=1}^n x_i p_i, \quad (5.1)$$

де x_i – значення випадкової величини;

p_i – ваговий коефіцієнт випадкової величини X .

Також треба розрахувати дисперсію.

Дисперсія – це міра розсіювання випадкової величини близько її математичного очікування. Вона дозволяє визначити віддаленість значень від їх середнього значення (математичного очікування).

Дисперсія виражена наступною формулою:

$$D[X] = \sum_{i=1}^n x_i^2 p_i - M[X]^2. \quad (5.2)$$

Обробка результатів виконується за допомогою порівняння математичного очікування та дисперсії кожного питання.

Розрахунки математичного очікування і дисперсії завдань були проведені за формулами (5.1– 5.2).

5.2 Розрахунки

5.2.1 Завдання 1

Під час виконання завдання 1, кожному учаснику необхідно було знайти інформацію на сайті за наступним питанням: «На якому курсі відбувається вибір спеціалізації навчання?»

Час проходження тестувань А і Б, завдання 1 приведений в (табл.5.3 – 5.4).

Таблиця 5.3 – Розподілу часу на виконання завдання 1 тестування А

x_{A_1i}	22.3	22.9	27.0	28.1	28.2	29.4	31.3	34.9	35.6	42.5	44.2	47.1	47.6	48.8	52.0	54.8	59.9	62.9	66.2
p_{A_1i}	0.05	0.05	0.05	0.05	0.1	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Таблиця 5.4 – Розподілу часу на виконання завдання 1 тестування Б

x_{B_1i}	5.8	6.3	7.7	7.9	8.1	9.2	11.3	11.8	12.5	12.9	15.2	15.6	16	16.1	16.3	17.1	18.5	19.4	19.7	20.2
p_{B_1i}	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Розрахунки математичного очікування та дисперсії:

$$\begin{aligned}
 M[A_1] = & 22.3 \times 0.05 + 22.9 \times 0.05 + 27.0 \times 0.05 + 28.1 \times 0.05 + 28.2 \times 0.1 \\
 & + 29.4 \times 0.05 + 31.3 \times 0.05 + 34.9 \times 0.05 + 35.6 \times 0.05 \\
 & + 42.5 \times 0.05 + 44.2 \times 0.05 + 47.1 \times 0.05 + 47.6 \times 0.05 \\
 & + 48.8 \times 0.05 + 52.0 \times 0.05 + 54.8 \times 0.05 + 59.9 \times 0.05 \\
 & + 62.9 \times 0.05 + 66.2 \times 0.05 = 40.7.
 \end{aligned}$$

$$\begin{aligned}
 D[A_1] = & 22.3^2 \times 0.05 + 22.9^2 \times 0.05 + 27.0^2 \times 0.05 + 28.1^2 \times 0.05 \\
 & + 28.2^2 \times 0.1 + 29.4^2 \times 0.05 + 31.3^2 \times 0.05 + 34.9^2 \times 0.05 \\
 & + 35.6^2 \times 0.05 + 42.5^2 \times 0.05 + 44.2^2 \times 0.05 + 47.1^2 \times 0.05 \\
 & + 47.6^2 \times 0.05 + 48.8^2 \times 0.05 + 52.0^2 \times 0.05 + 54.8^2 \times 0.05 \\
 & + 59.9^2 \times 0.05 + 62.9^2 \times 0.05 + 66.2^2 \times 0.05 - 40.7^2 = 180.
 \end{aligned}$$

$$\begin{aligned}
 M[B_1] = & 5.8 \times 0.05 + 6.3 \times 0.05 + 7.7 \times 0.05 + 7.9 \times 0.05 + 8.1 \times 0.05 \\
 & + 9.2 \times 0.05 + 11.3 \times 0.05 + 11.8 \times 0.05 + 12.5 \times 0.05 \\
 & + 12.9 \times 0.05 + 15.2 \times 0.05 + 15.6 \times 0.05 + 16 \times 0.05 \\
 & + 16.1 \times 0.05 + 16.3 \times 0.05 + 17.1 \times 0.05 + 18.5 \times 0.05 \\
 & + 19.4 \times 0.05 + 19.7 \times 0.05 + 20.2 \times 0.05 = 13.38.
 \end{aligned}$$

$$\begin{aligned}
D[B_1] &= 5.8^2 \times 0.05 + 6.3^2 \times 0.05 + 7.7^2 \times 0.05 + 7.9^2 \times 0.05 \\
&+ 8.1^2 \times 0.05 + 9.2^2 \times 0.05 + 11.3^2 \times 0.05 + 11.8^2 \times 0.05 \\
&+ 12.5^2 \times 0.05 + 12.9^2 \times 0.05 + 15.2^2 \times 0.05 + 15.6^2 \times 0.05 \\
&+ 16^2 \times 0.05 + 16.1^2 \times 0.05 + 16.3^2 \times 0.05 + 17.1^2 \times 0.05 \\
&+ 18.5^2 \times 0.05 + 19.4^2 \times 0.05 + 19.7^2 \times 0.05 + 20.2^2 \times 0.05 \\
&= 20.8.
\end{aligned}$$

Різниця математичного очікування завдання 1, тестів А і Б, скала 27.3. Це свідчить про те, що пошук відповіді на перше запитання з використанням чат-боту був швидше на 27.3 секунди ніж вручну.

5.2.2 Завдання 2

Завдання 2 передбачало пошук інформації на питання: «Треба знайти основну інформацію про предмет Основи розробки UI/UX»

Час проходження тестувань завдання 2 тесту А і тесту Б приведений в (табл. 5.5 – 5.6).

Таблиця 5.5 – Розподілу часу на виконання завдання 2 тестування А

x_{A2i}	24.7	31.4	34.3	34.8	36.4	37.1	41.0	41.1	41.3	50.6	51.0	52.7	53.5	53.6	59.1	59.8	59.9	60.3	60.7
p_{A2i}	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.1	0.05	0.05	0.05

Таблиця 5.6 – Розподілу часу на виконання завдання 2 тестування Б

x_{B1i}	3	4.3	5.7	7.6	12.2	12.3	13.4	14.1	14.6	15	15.3	15.5	16.8	17.3	17.5	18.2	19.2	21.1	23.1	27.3
p_{B2i}	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Розрахунки математичного очікування та дисперсії:

$$\begin{aligned}
M[A_2] = & 24.7 \times 0.05 + 31.4 \times 0.05 + 34.3 \times 0.05 + 34.8 \times 0.05 + 36.4 \times 0.05 \\
& + 37.1 \times 0.05 + 41.0 \times 0.05 + 41.1 \times 0.05 + 41.3 \times 0.05 \\
& + 50.6 \times 0.05 + 51.0 \times 0.05 + 52.7 \times 0.05 + 53.5 \times 0.05 \\
& + 53.6 \times 0.05 + 59.1 \times 0.05 + 59.8 \times 0.1 + 59.9 \times 0.05 \\
& + 60.3 \times 0.05 + 60.7 \times 0.05 = 47.2
\end{aligned}$$

$$\begin{aligned}
D[A_2] = & 24.7^2 \times 0.05 + 31.4^2 \times 0.05 + 34.3^2 \times 0.05 + 34.8^2 \times 0.05 \\
& + 36.4^2 \times 0.05 + 37.1^2 \times 0.05 + 41.0^2 \times 0.05 + 41.1^2 \times 0.05 \\
& + 41.3^2 \times 0.05 + 50.6^2 \times 0.05 + 51.0^2 \times 0.05 + 52.7^2 \times 0.05 \\
& + 53.5^2 \times 0.05 + 53.6^2 \times 0.05 + 59.1^2 \times 0.05 + 59.8^2 \times 0.1 \\
& + 59.9^2 \times 0.05 + 60.3^2 \times 0.05 + 60.7^2 \times 0.05 - 47.2^2 = 125.9.
\end{aligned}$$

$$\begin{aligned}
M[B_2] = & 3 \times 0.05 + 4.3 \times 0.05 + 5.7 \times 0.05 + 7.6 \times 0.05 + 12.2 \times 0.05 \\
& + 12.3 \times 0.05 + 13.4 \times 0.05 + 14.1 \times 0.05 + 14.6 \times 0.05 \\
& + 15 \times 0.05 + 15.3 \times 0.05 + 15.5 \times 0.05 + 16.8 \times 0.05 \\
& + 17.3 \times 0.05 + 17.5 \times 0.05 + 18.2 \times 0.05 + 19.2 \times 0.05 \\
& + 21.1 \times 0.05 + 23.1 \times 0.05 + 27.3 \times 0.05 = 14.7.
\end{aligned}$$

$$\begin{aligned}
D[B_2] = & 3^2 \times 0.05 + 4.3^2 \times 0.05 + 5.7^2 \times 0.05 + 7.6^2 \times 0.05 + 12.2^2 \times 0.05 \\
& + 12.3^2 \times 0.05 + 13.4^2 \times 0.05 + 14.1^2 \times 0.05 + 14.6^2 \times 0.05 \\
& + 15^2 \times 0.05 + 15.3^2 \times 0.05 + 15.5^2 \times 0.05 + 16.8^2 \times 0.05 \\
& + 17.3^2 \times 0.05 + 17.5^2 \times 0.05 + 18.2^2 \times 0.05 + 19.2^2 \times 0.05 \\
& + 21.1^2 \times 0.05 + 23.1^2 \times 0.05 + 27.3^2 \times 0.05 - 14.7^2 = 35.7
\end{aligned}$$

Різниця математичного очікування завдання 2, тестів А і Б, скала 32.5. Це свідчить про те, що пошук відповіді на перше запитання з використанням чат-боту був швидше на 32.5 секунди ніж вручну.

5.2.3 Завдання 3

Завдання 3 полягало в пошуку інформації про ціну контрактного навчання: «Скільки коштує контрактне навчання на бакалавраті та магістратурі?»

Виходячи з результатів часу виконання завдання 3 побудовані ряди розподілу величин для тесту А і тесту Б (табл. 5.7 – 5.8).

Таблиця 5.7 – Розподілу часу на виконання завдання 3 тестування А

x_{A3i}	21.3	21.8	22.4	34.6	36.0	42.2	43.1	43.8	44.4	50.1	51.9	53.0	55.0	56.9	57.2	57.4	60.7	62.5	68.0
p_{A3i}	0.05	0.05	0.05	0.05	0.05	0.05	0.1	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Таблиця 5.8 – Розподілу часу на виконання завдання 3 тестування Б

x_{B3i}	2.4	3.2	3.8	3.9	5.1	5.3	5.6	6.2	6.3	6.4	7.8	8	9.2	10.1	10.4	10.8	11.4	12.6	15.8
p_{B3i}	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.1	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Розрахунки математичного очікування та дисперсії:

$$\begin{aligned}
 M[A_3] = & 21.3 \times 0.05 + 21.8 \times 0.05 + 22.4 \times 0.05 + 34.6 \times 0.05 + 36.0 \times 0.05 \\
 & + 42.2 \times 0.05 + 43.1 \times 0.05 + 43.8 \times 0.05 + 44.4 \times 0.05 \\
 & + 50.1 \times 0.05 + 51.9 \times 0.05 + 53.0 \times 0.05 + 55.0 \times 0.05 \\
 & + 56.9 \times 0.05 + 57.2 \times 0.05 + 57.4 \times 0.05 + 60.7 \times 0.05 \\
 & + 62.5 \times 0.05 + 68.0 \times 0.05 = 46.3
 \end{aligned}$$

$$\begin{aligned}
 D[A_3] = & 21.3^2 \times 0.05 + 21.8^2 \times 0.05 + 22.4^2 \times 0.05 + 34.6^2 \times 0.05 \\
 & + 36.0^2 \times 0.05 + 42.2^2 \times 0.05 + 43.1^2 \times 0.05 + 43.8^2 \times 0.05 \\
 & + 44.4^2 \times 0.05 + 50.1^2 \times 0.05 + 51.9^2 \times 0.05 + 53.0^2 \times 0.05 \\
 & + 55.0^2 \times 0.05 + 56.9^2 \times 0.05 + 57.2^2 \times 0.05 + 57.4^2 \times 0.05 \\
 & + 60.7^2 \times 0.05 + 62.5^2 \times 0.05 + 68.0^2 \times 0.05 - 46.3^2 = 177.3.
 \end{aligned}$$

Таблиця 5.10 – Розподілу часу на виконання завдання 4 тестування Б

x_{B_4i}	2.5	2.8	3.6	4.2	5.2	5.7	6	6.1	7.4	8	9	9.8	10.7	11.1	11.3	13.5	13.6
p_{B_4i}	0.05	0.1	0.05	0.05	0.05	0.05	0.05	0.1	0.05	0.05	0.05	0.05	0.1	0.05	0.05	0.05	0.05

Розрахунки математичного очікування та дисперсії:

$$\begin{aligned}
 M[A_4] = & 21.0 \times 0.05 + 21.4 \times 0.05 + 27.2 \times 0.05 + 28.9 \times 0.05 + 31.0 \times 0.05 \\
 & + 36.3 \times 0.05 + 36.4 \times 0.05 + 41.0 \times 0.05 + 42.8 \times 0.05 \\
 & + 44.7 \times 0.05 + 46.7 \times 0.05 + 49.1 \times 0.05 + 49.9 \times 0.05 \\
 & + 54.2 \times 0.05 + 56.6 \times 0.05 + 61.8 \times 0.05 + 64.4 \times 0.05 \\
 & + 65.5 \times 0.05 + 70.1 \times 0.05 = 46.
 \end{aligned}$$

$$\begin{aligned}
 D[A_4] = & 21.0^2 \times 0.05 + 21.4^2 \times 0.05 + 27.2^2 \times 0.05 + 28.9^2 \times 0.05 \\
 & + 31.0^2 \times 0.05 + 36.3^2 \times 0.05 + 36.4^2 \times 0.05 + 41.0^2 \times 0.05 \\
 & + 42.8^2 \times 0.05 + 44.7^2 \times 0.05 + 46.7^2 \times 0.05 + 49.1^2 \times 0.05 \\
 & + 49.9^2 \times 0.05 + 54.2^2 \times 0.05 + 56.6^2 \times 0.05 + 61.8^2 \times 0.05 \\
 & + 64.4^2 \times 0.05 + 65.5^2 \times 0.05 + 70.1^2 \times 0.05 - 46^2 = 233.1.
 \end{aligned}$$

$$\begin{aligned}
 M[B_4] = & 2.5 \times 0.05 + 2.8 \times 0.05 + 3.6 \times 0.05 + 4.2 \times 0.05 + 5.2 \times 0.05 \\
 & + 5.7 \times 0.05 + 6 \times 0.05 + 6.1 \times 0.05 + 7.4 \times 0.05 + 8 \times 0.05 \\
 & + 9 \times 0.05 + 9.8 \times 0.05 + 10.7 \times 0.05 + 11.1 \times 0.05 \\
 & + 11.3 \times 0.05 + 13.5 \times 0.05 + 13.6 \times 0.05 = 7.5
 \end{aligned}$$

$$\begin{aligned}
 D[B_4] = & 2.5^2 \times 0.05 + 2.8^2 \times 0.05 + 3.6^2 \times 0.05 + 4.2^2 \times 0.05 + 5.2^2 \times 0.05 \\
 & + 5.7^2 \times 0.05 + 6^2 \times 0.05 + 6.1^2 \times 0.05 + 7.4^2 \times 0.05 + 8^2 \times 0.05 \\
 & + 9^2 \times 0.05 + 9.8^2 \times 0.05 + 10.7^2 \times 0.05 + 11.1^2 \times 0.05 \\
 & + 11.3^2 \times 0.05 + 13.5^2 \times 0.05 + 13.6^2 \times 0.05 - 7.5^2 = 9.9.
 \end{aligned}$$

Різниця математичного очікування завдання 4, тестів А і Б, скала 38.5. Це свідчить про те, що пошук відповіді на перше запитання з використанням чат-боту був швидше на 38.5 секунди ніж вручну.

5.2.5 Завдання 5

Завдання 5 полягає в пошуку «Які основні навчальні дисципліни бакалавр»

Виходячи з результатів часу виконання завдання 5 побудовані ряди розподілу величин для тестування А і тестування Б (табл. 5.11– 5.12).

Таблиця 5.11 – Розподілу часу на виконання завдання 5 тестування А

x_{A5_i}	21.6	27.5	29.8	35.7	42.2	42.8	45.1	46.5	46.9	48.8	49.5	61.0	61.9	62.6	63.2	63.9	64.9	66.4	68.6	70.5
p_{A5_i}	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Таблиця 5.12 – Розподілу часу на виконання завдання 5 тестування Б

x_{B5_i}	3.9	5.2	5.3	5.5	6.4	7	7.7	8.4	8.6	9.1	9.7	10.2	11	11.7	12.2	12.4	14.5	15.3	15.8	17.8
p_{B5_i}	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

Розрахунки математичного очікування та дисперсії:

$$\begin{aligned}
 M[A_5] = & 21.6 \times 0.05 + 27.5 \times 0.05 + 29.8 \times 0.05 + 35.7 \times 0.05 + 42.2 \times 0.05 \\
 & + 42.8 \times 0.05 + 45.1 \times 0.05 + 46.5 \times 0.05 + 46.9 \times 0.05 \\
 & + 48.8 \times 0.05 + 49.5 \times 0.05 + 61.0 \times 0.05 + 61.9 \times 0.05 \\
 & + 62.6 \times 0.05 + 63.2 \times 0.05 + 63.9 \times 0.05 + 64.9 \times 0.05 \\
 & + 66.4 \times 0.05 + 68.6 \times 0.05 + 70.5 \times 0.05 = 51
 \end{aligned}$$

$$\begin{aligned}
D[A_5] = & 21.6^2 \times 0.05 + 27.5^2 \times 0.05 + 29.8^2 \times 0.05 + 35.7^2 \times 0.05 \\
& + 42.2^2 \times 0.05 + 42.8^2 \times 0.05 + 45.1^2 \times 0.05 + 46.5^2 \times 0.05 \\
& + 46.9^2 \times 0.05 + 48.8^2 \times 0.05 + 49.5^2 \times 0.05 + 61.0^2 \times 0.05 \\
& + 61.9^2 \times 0.05 + 62.6^2 \times 0.05 + 63.2^2 \times 0.05 + 63.9^2 \times 0.05 \\
& + 64.9^2 \times 0.05 + 66.4^2 \times 0.05 + 68.6^2 \times 0.05 + 70.5^2 \times 0.05 \\
& - 51^2 = 205.7.
\end{aligned}$$

$$\begin{aligned}
M[B_5] = & 3.9 \times 0.05 + 5.2 \times 0.05 + 5.3 \times 0.05 + 5.5 \times 0.05 + 6.4 \times 0.05 \\
& + 7 \times 0.05 + 7.7 \times 0.05 + 8.4 \times 0.05 + 8.6 \times 0.05 + 9.1 \times 0.05 \\
& + 9.7 \times 0.05 + 10.2 \times 0.05 + 11 \times 0.05 + 11.7 \times 0.05 \\
& + 12.2 \times 0.05 + 12.4 \times 0.05 + 14.5 \times 0.05 + 15.3 \times 0.05 \\
& + 15.8 \times 0.05 + 17.8 \times 0.05 = 9.9.
\end{aligned}$$

$$\begin{aligned}
D[B_5] = & 3.9^2 \times 0.05 + 5.2^2 \times 0.05 + 5.3^2 \times 0.05 + 5.5^2 \times 0.05 + 6.4^2 \times 0.05 \\
& + 7^2 \times 0.05 + 7.7^2 \times 0.05 + 8.4^2 \times 0.05 + 8.6^2 \times 0.05 \\
& + 9.1^2 \times 0.05 + 9.7^2 \times 0.05 + 10.2^2 \times 0.05 + 11^2 \times 0.05 \\
& + 11.7^2 \times 0.05 + 12.2^2 \times 0.05 + 12.4^2 \times 0.05 + 14.5^2 \times 0.05 \\
& + 15.3^2 \times 0.05 + 15.8^2 \times 0.05 + 17.8^2 \times 0.05 - 9.9^2 = 14.6.
\end{aligned}$$

Різниця математичного очікування завдання 5, тестів А і Б, скала 41.1. Це свідчить про те, що пошук відповіді на перше запитання з використанням чат-боту був швидше на 41.1 секунди ніж вручну.

5.2.6 Об'єднання результатів тестувань

Результати виконаних розрахунків для більшої наявності об'єднано та згруповано в табл. 5.13.

Таблиця 5.13 – Результати виконаних розрахунків.

Параметр	Номер завдання				
	1	2	3	4	5
$M[A_{1...5}]$	40.7	47.2	46.3	46.0	51.0
$D[A_{1...5}]$	180.0	125.9	177.3	233.1	205.7
$M[B_{1...5}]$	13.4	14.7	7.5	7.5	9.9
$D[B_{1...5}]$	20.8	35.7	11.5	11.9	14.6

5.2.7 Розрахунок загального математичного очікування та міжгрупової дисперсії

На основі проведених розрахунків для кожного завдання проводиться визначення загального математичного очікування кожної групи за формулою (5.1).

$$M[A] = \frac{40.7 + 47.2 + 46.3 + 46 + 51}{5} = 46.2$$

$$M[B] = \frac{13.4 + 14.7 + 7.5 + 7.5 + 9.9}{5} = 10.6$$

Також розраховується міжгрупова дисперсія, яка характеризує систематичну варіацію результативної ознаки, яка обумовлена впливом ознаки-фактора, покладеного в основу угруповання, тобто варіацію часу виконання завдань групою. Розраховується за формулою:

$$\delta^2 = \frac{\sum_{j=1}^n (\bar{x}_j - \bar{x})^2 \times n_j}{\sum_{j=1}^p n_j}, \quad (5.3)$$

де n_j — чисельність одиниць у групі;

j – номер групи;

\bar{x}_j – середнє значення ознаки в j групі;

\bar{x} – загальна середня, відповідна групового математичного очікування.

Міжгрупова дисперсія розраховується по кожній групі:

$$\begin{aligned} \delta_A^2 &= ((40.7 - 46.2)^2 + (47.2 - 46.2)^2 + (46.3 - 46.2)^2 + (46 - 46.2)^2 \\ &\quad + (11 - 46.2)^2)/5 = \frac{53.9}{5} = 10.8 \end{aligned}$$

$$\begin{aligned} \delta_B^2 &= ((13.4 - 10.6)^2 + (14.7 - 10.6)^2 + (7.5 - 10.6)^2 + (7.5 - 10.6)^2 \\ &\quad + (9.9 - 10.6)^2)/5 = \frac{43.9}{5} = 8.8 \end{aligned}$$

Таким чином після проведення тестувань та аналізу даних були отримані результати, на основі котрих було розраховано математичне очікування часу виконання всіх завдань і міжгрупову дисперсію, що відражає відхилення від середнього часу виконання завдань.

Можна зробити висновок, що час виконання завдань тестування А більше, ніж тестування Б на 35.6 секунд. Це говорить, про те, що впровадження чат-боту на сайт кафедри МСТ скоротило час роботи користувачів на 77.1%.

6 ЕКОНОМІЧНА ЧАСТИНА

6.1 Характеристика науково-дослідної роботи

В економічній частині атестаційної роботи обґрунтовано економічну доцільність дослідження та розробки методики ефективного використання чат-боту у ВЕБ, на прикладі сайту кафедри МСТ.

У роботі проведено дослідження актуальної інформації щодо чат-ботів: їх видів, структур, принципів роботи та характеристик чат-ботів. На основі отриманих даних розроблена методика щодо ефективного використання чат-ботів у ВЕБ.

Реалізація розробленої методики дозволяє:

- значно скоротити час пошуку інформації на сайтах;
- підвищити влучність відповідей на запитання користувачів;
- зменшити час, який використовує розробник на створення чат-боту.

6.2 Етапи виконання НДР, їх трудомісткість та заробітна плата

У процесі виконання роботи був проведений огляд існуючих тенденцій у ВЕБ, проведено дослідження принципів роботи чат-ботів, їх види та структура, основи QA-систем, штучного інтелекту, на основі аналізу спеціальної літератури розглянуті принципи, за якими працюють чат-боти.

На основі даних дослідження розроблена методика, яка підвищує ефективність використання чат-ботів у ВЕБ.

На основі запропонованої методики було розроблено чат-бот кафедри МСТ.

Після цього проведено експеримент, у результаті якого оцінено ефективність розробленої методики.

Умовно науково-дослідну роботу (НДР) можна розділити на такі етапи: підготовчий, основний і заключний.

На стадії виконання підготовчого етапу проаналізовано актуальність даної роботи. Здійснено підбір документації щодо чат-ботів, штучного інтелекту тощо.

На етапі виконання основної частини НДР були виконані такі роботи:

- проведено дослідження;
- на основі дослідження розроблено методику;
- відповідно до запропонованої методики розроблено чат-бот;
- проведено експеримент та оброблено його результати.

У заключній частині здійснюється оцінка ефективності виконання НДР, складання звіту по НДР, захист звіту.

Найбільш складною й відповідальною частиною при плануванні НДР є розрахунок трудомісткості робіт, тому що трудові витрати часто становлять основну частину вартості науково-дослідних робіт і безпосередньо впливають на строки розробки.

Для виконання роботи було залучено 29 осіб, контролювали процеси керівник роботи та керівник тестування. Загальна чисельність робочої групи виконавців НДР склала 29 осіб. Групу найнятої робочої сили склали:

- керівник роботи – 1 особа, заробітна плата 8 492 грн./міс.;
- керівник тестування – 1 особа, заробітна плата 5 000 грн./міс.;
- учасники тестування – 20 осіб, заробітна плата 5 000 грн./міс.;
- спеціаліст тестування – 1 особа, заробітна плата 5 000 грн./міс.;
- дизайнер інтерфейсів – 1 особа, заробітна плата 34 134 грн./міс.;
- веб-аналітик – 1 особа, заробітна плата 24 445 грн./міс.;
- текстовий редактор – 1 особа, заробітна плата 18 000 грн./міс.;
- програміст – 1 особа, заробітна плата 19 912 грн./міс.;
- верстальник – 1 особа, заробітна плата 18 489 грн./міс.;
- науковий співробітник – 1 особа, заробітна плата 9 477 грн./міс.

Всі заробітні плати актуальні на листопад 2020 року. Джерело – «dou.ua».

Проведемо розрахунок трудовитрат і заробітної плати виконавців робіт.

Середньоденна заробітна плата виконавця робіт ($Z_{\text{ср.дн.}}$) розраховується за формулою:

$$Z_{\text{ср.дн.}} = \frac{Z_{\text{ср.міс.}}}{n}, \quad (6.1)$$

де $Z_{\text{ср.міс.}}$ – середньомісячна зарплата виконавця роботи;

n – число робочих днів у місяці, ($n = 22$).

Етапи виконання НДР, перелік і зміст робіт, трудомісткість їх виконання, заробітна плата виконавців робіт представлені в (табл. 6.1).

Таблиця 6.1 – Розрахунок трудовитрат і заробітної плати виконавця робіт

Перелік робіт	Кількість виконавців	Посада виконавця	Трудомісткість робіт, люд.-днів	Середньоденна заробітна плата, грн.	Сума заробітної плати, грн.
1	2	3	4	5	6
1. Підготовчий етап					
1.1 Розробка та затвердження ТЗ	1	Керівник роботи	2	386,00	772,00
1.2 Підготовка довідкових матеріалів та даних для виконання НДР	1	Керівник роботи	3	386,00	1158,00
2. Основний етап					
2.1 Постановка задачі	1	Керівник роботи	1	386	386
2.2 проведення дослідження матеріалу	1	Науковий співробітник	1	430,77	430,77
2.3 Розробка методики	1	Науковий співробітник	1	430,77	430,77
2.4 Розробка чат-боту					
2.4.1 Веб-анлітик збирає статистику сайту	1	Веб-анлітик	1	1111,13	1111,13

Продовження таблиці 6.1

1	2	3	4	5	6
2.4.2 Складання текстової БД питань/відповідей	1	Текстовий редактор	10	818,18	8181,81
2.4.3 Розробка чат-боту	1	Програміст	10	905,09	9050,90
2.4.4 Розробка візуального оформлення	1	Дизайнер	1	1551,54	1551,54
2.4.5 Верстання чат-боту	1	Верстальник	1	840,40	840,40
2.5 Проведення тестування	1	Керівник тестування	1	227,27	227,27
2.5.1 Група тестувальників	20	Учасники	1	227,27	4545,45
2.6 Обробка результатів експерименту	1	Спеціаліст тестування	2	227,27	454,54
2.7 Оцінка розрахованих даних експерименту	1	Спеціаліст тестування	1	227,27	227,27
3. Заключний етап					
3.1 Технічне оформлення звіту про виконання НДР	1	Керівник роботи	2	386	772
3.2 Оцінка виконання науково-дослідної роботи	1	Керівник роботи	2	386	772
Всього			40	8926,96	26 593,66

6.3 Розрахунок одноразових витрат на розробку НДР

Калькуляція собівартості розраховується відповідно до існуючих нормативних актів України. До складу калькуляції входять такі статті витрат:

- матеріальні витрати;
- витрати на оплату праці;
- єдиний соціальний внесок;
- амортизація основних засобів (вартість машинного часу);
- витрати на спожиту електроенергію;

– інші витрати.

До інших витрат відносяться адміністративні витрати (водопостачання, водовідведення, опалення, освітлення) та вартість послуг зв'язку.

Матеріальні витрати визначаються витратами на матеріали, визначені їх потребою для виконання робіт, і цін, що діють на момент складання калькуляції.

Матеріальні витрати розраховуються за такою формулою:

$$M = \sum_{j=1}^n Q_j \times C_j, \quad (6.2)$$

де M – сумарні витрати на матеріали, в тому числі малоцінні предмети, що швидко зношуються (носії, папір, канцелярське приладдя тощо), або на літературу, яка необхідна для проведення роботи, тощо;

Q_j – кількість використаних одиниць j -го виду матеріалів, $j = (1 \div n)$;

C_j – ціна одиниці j -го виду матеріалів.

Розрахунок матеріальних витрат представлено в (табл. 6.2).

Таблиця 6.2 – Розрахунок матеріальних витрат

Найменування	Од. вим.	Кількість од.	Ціна, грн	Сума, грн.
Олівець механічний	уп.	2	60,00	120,00
Ручки	уп.	2	50	100,00
Папір	уп.	2	80,00	160,00
Степлер	шт.	15	17,00	255,00
Скріпки для степлеру	уп.	2	7,00	14,00
Диск	шт.	3	7,00	21,00
Заправка для картриджу	шт.	4	180,00	720,00
Калькулятор	шт.	4	37,00	148,00
Всього				1538

Витрати на оплату праці розраховуються виходячи з необхідного для виконання робіт складу й кількості працівників, а також із середньомісячної

заробітної плати. Відповідно до проведених розрахунків витрати на оплату праці виконавців роботи дорівнюють 26593,66 грн.

Єдиний внесок на загальнодержавне соціальне страхування (ЄСВ) – консолідований страховий внесок, збір якого здійснюється в систему загальнообов’язкового державного соціального страхування в обов’язковому порядку і на регулярній основі з метою забезпечення захисту у випадках, передбачених законодавством, прав застрахованих осіб і членів їх сімей на отримання страхових виплат (послуг) за діючими видами загальнообов’язкового державного соціального страхування.

Для об’єкта дослідження ставка єдиного соціального внеску дорівнює 22 % від витрат на оплату праці, тобто розмір ЄСВ дорівнює 5850,60 грн.

При виконанні НДР застосовувалось наступне обладнання: комп’ютери вартістю 10000 грн., 8 шт. та принтери вартістю 2700 грн., 4 шт.

Вищенаведене устаткування є власністю організації виконавця, тому доцільно розрахувати суму амортизаційних відрахувань на період виконання НДР. Амортизація основних засобів розраховується за формулою:

$$AB = \sum_{k=1}^L \frac{BO_k}{TE_k} \times T, \quad (6.3)$$

де AB – сума амортизаційних відрахувань, нарахованих під час проведення науково-дослідницької роботи;

BO_k – вартість основних засобів k -го виду;

TE_k – термін експлуатації основних засобів k -го виду, днів;

T – термін науково-дослідницької роботи, днів;

L – кількість видів обладнання.

Для даного випадку, термін експлуатації основних засобів складає 3 роки або 750 робочих днів.

Підставивши відомі значення у (6.3), визначимо величину амортизаційних відрахувань:

$$AB = \left(\frac{10000 \times 8}{750} + \frac{2700 \times 4}{750} \right) \times 40 = 4842,4 \text{ грн.}$$

Витрати на використану обладнанням електроенергію розраховуються за формулою:

$$Z_e = M \cdot t \cdot T_{кВт}, \quad (6.4)$$

де M – потужність устаткування, тобто кількість енергії, споживаної за одиницю часу (кВт/година);

t – кількість годин використання устаткування за період проведення науково-дослідницької роботи;

$T_{кВт}$ – тариф, тобто вартість використання 1 кВт електроенергії.

Споживна потужність комп'ютера складає 0,7 кВт та принтера 0,8 кВт за годину. Тариф споживачів за першим класом напруги, за версією сайта «Ахіомplus», тобто 27,5 кВт та більше), складає 1,8065 грн./кВтгодин (без ПДВ). Підставивши значення у формулу (5.4), визначимо величину витрат на спожиту електроенергію:

$$Z_e = (0,5 \times 8 \times 320 + 0,8 \times 4 \times 5) \times 1,8065 = 2341,22 \text{ грн.}$$

До інших статей витрат відносяться такі:

- адміністративні витрати: (водопостачання, водовідведення, освітлення, опалення), які прийнято у розмірі 20% від витрат на оплату праці;
- вартість оплати послуг зв'язку.

Вартість оплати послуг зв'язку становитиме:

а) Інтернет – із розрахунку 200 грн. на місяць (безлімітний пакет); всього 400 грн. за 40 дні виконання НДР;

б) телефон – із розрахунку 120 грн на місяць; всього 240 грн. за 40 днів.

За час виконання НДР витрати на відрядження, аутсорсинг, інформаційні послуги та маркетингові заходи не мали місця.

Для виконання НДР використовувалося безкоштовне ПО, наприклад, для розробки чат-боту Dialogflow, та для прототипування – Figma.

Результати розрахунку кошторису витрат, тобто одноразових витрат, на виконання НДР наведені в табл. 6.3.

Таблиця 6.3 – Кошторис витрат на розробку НДР

№ з/п	Стаття витрат	Сума, грн.
1	Заробітна плата	26593,66
2	Єдиний соціальний внесок (22,0 % від п.1)	5850,60
3	Матеріальні витрати	1538
4	Амортизація основних засобів	4842,4
5	Витрати на спожиту електроенергію	2341,22
6	Інші витрати, у тому числі:	–
6.1	Адміністративні витрати (20,0 % від п.1)	5318,73
6.2	Вартість послуг зв'язку	640,00
7	Всього витрати	47124,61

Таким чином, кошторис витрат на виконання даної НДР відбиває сумарні витрати за статтями складає 47124.61 грн.

6.4 Оцінка результатів науково-дослідної роботи

Результат – це наслідок послідовності дій виконаних при НДР, виражений якісно або кількісно. В загальному випадку оцінка результатів НДР – це визначення ефективності отриманих рішень порівняно з сучасним науково-технічним рівнем.

Відповідно до теми даної атестаційної роботи можна зробити висновок про те, що у якості результату впровадження НДР є зменшення часу на досягнення мети користувача (реєстрація дитини в ЗДО).

Результат від впровадження НДР визначається за такою формулою:

$$\Delta P_j = |X_{бj} - X_{нj}|, \quad (6.5)$$

де ΔP_j – покращення j -ої характеристики досліджуваного процесу за рахунок впровадження результатів НДР ($j=1,m$);

m – кількість досліджуваних характеристик;

$X_{\bar{j}}$ – базове значення j -ої характеристики;

X_{n_j} – нове значення j -ої характеристики після впровадження НДР.

У якості досліджуваної характеристики виступає час, використаний на досягнення мети користувачем (пошук інформації на сайті кафедри МСТ).

Пошук здійснювався двома способами:

– простим, ручним способом;

– за допомогою чат-боту.

На основі даних отриманих під час проведення експерименту, були розраховані міжгрупове математичне очікування та дисперсія.

Математичне очікування – середній час виконання всіх завдань.

Отримані результати тестування наведені у табл. 6.4.

Таблиця 6.4 – Час досягнення мети користувачем

Тип пошуку	Математичне очікування – середній час виконання всіх завдань (сек)
Ручний пошук	46,2
Пошук з чат-ботом	10,6

6.5 Визначення економічної ефективності результатів НДР

Для визначення економічної ефективності результатів НДР необхідно порівняти витрати на розробку НДР з отриманими результатами.

Основним показником економічної ефективності науково-дослідної роботи є коефіцієнт «ефект-витрати», який розраховується за формулою:

$$K_{ee} = \frac{\Delta P_j}{B_p}, \quad (6.6)$$

B_p – витрати (кошторисна вартість) на виконання НДР, грн.;

де $K_{ев}$ – коефіцієнт «ефект-витрати», який відбиває, наскільки кожна гривня витрат НДР змінює j -ту характеристику досліджуваного процесу.

Підставивши раніше визначені значення до (6.6), розрахуємо чисельне значення коефіцієнту «ефект-витрати»:

$$K_{ев} = \frac{35,6}{47124,61} \cdot 100\% = 0,07554 (\%).$$

У результаті проведених досліджень, можна зробити висновок про те, що кожна гривня витрат на розробку НДР забезпечує зниження затрат часу на пошук необхідної інформації використовуючи чат-бота на сайті кафедри МСТ на 0,07554%. Дана науково-дослідна робота має позитивний показник економічної ефективності. Роботу у цілому можна враховувати ефективною або такою, що має науковий та технічний рівень.

ВИСНОВКИ

У результаті виконання атестаційної роботи магістра було здійснено дослідження видів, структур, принципів роботи та характеристик чат-ботів. На основі проведеного дослідження було розроблено методику ефективного використання чат-ботів у WEB, що дозволяє значно скоротити час пошуку інформації на сайтах. Методика враховує, у тому числі, й дані Google Analytics, які дозволяють відстежити поведінку користувача на сайті, а саме дослідити найбільш відвідувані сторінки та найчастіші пошукові запити. Такий підхід дозволяє здійснити «тонке налаштування» проектованого чат-боту. В роботі було здійснено практичну реалізацію чат-боту із урахуванням розробленої методики. Використання методів математичної статистики дозволило визначити основні математичні характеристики, що доводять ефективність та доцільність використання розробленої методики. В роботі здійснено економічне обґрунтування ефективності застосування запропонованої методики.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Lara C. Top Web Design Trends for 2021 [Електронний ресурс] / Christopher Lara // theedigital. – 2020. – Режим доступу до ресурсу: <https://www.theedigital.com/blog/web-design-trends>.
2. OISIN M. E-BOOK CHATBOTS: An Introduction And Easy Guide To Making Your Own / Muldowney OISIN. – Дублін, Ірландія: Curces & Magic, 2017. – 76 с.
3. Dorothy C. The Story of ELIZA, the First Chatbot Developed In The 1960s [Електронний ресурс] / Crotts Dorothy // journalismai.com. – 2019. – Режим доступу до ресурсу: <https://journalismai.com/2019/09/12/the-story-of-eliza-the-first-chatbot-developed-in-the-1960s-mr-journalism/>.
4. A brief history of Chatbots [Електронний ресурс] // chatbotslife.com. – 2019. – Режим доступу до ресурсу: <https://chatbotslife.com/a-brief-history-of-chatbots-d5a8689cf52f>.
5. Frankenfield J. Turing Test [Електронний ресурс] / Jake Frankenfield // investopedia.com. – 2020. – Режим доступу до ресурсу: <https://www.investopedia.com/terms/t/turing-test.asp#:~:text=What%20Is%20the%20Turing%20Test,it%20has%20demonstrated%20human%20intelligence>.
6. Frankenfield J. Artificial Intelligence (AI) [Електронний ресурс] / Jake Frankenfield // investopedia.com. – 2020. – Режим доступу до ресурсу: <https://www.investopedia.com/terms/a/artificial-intelligence-ai.asp>.
7. Dialogflow [Електронний ресурс] // Wikipedia. – 2020. – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/Dialogflow>.
8. Dialogflow ES basics [Електронний ресурс] // Google cloud. – 2020. – Режим доступу до ресурсу: <https://cloud.google.com/dialogflow/es/docs/basics>.
9. The Other Side of Chatbots: Disadvantages and Risks [Електронний ресурс] // aivo. – 2019. – Режим доступу до ресурсу: <https://www.aivo.co/blog/advantages-and-disadvantages-of-chatbots>.

10. Chatbot [Электронний ресурс] // Wikipedia. – 2020. – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/Chatbot>.
11. Natural language question answering: the view from here / L. HIRSCHMAN, R. GAIZAUSKAS. – Cambridge, England: Cambridge University Press, 2020. – 300 с. – (Cambridge University Press).
12. Issues, Tasks and Program Structures to Roadmap Research in Question Answering (QA) / [J. Burger, C. Israel, V. Chaundhri та ін.]. – Cambridge, England: Cambridge University Press, 2003. – 300 с.
13. Natural-language processing, NLP [Электронний ресурс] // wikipedia. – 2020. – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/%D0%9E%D0%B1%D1%80%D0%BE%D0%B1%D0%BA%D0%B0_%D0%BF%D1%80%D0%B8%D1%80%D0%BE%D0%B4%D0%BD%D0%BE%D1%97_%D0%BC%D0%BE%D0%B2%D0%B8.
14. Berry M. W. Survey of Text Mining I: Clustering, Classification, and Retrieval / Michael W. Berry. – Knoxville: Department of Computer Science University of Tennessee, 2003. – 241 с.
15. FREITAG D. Machine Learning for Information Extraction in Informal Domains / DAYNE FREITAG. – Pittsburgh: Kluwer Academic Publisher, 2000. – 202 с. – (Justsystem Pittsburgh Research Center).
16. Bizer C. Linked Data - The Story So Far [Электронний ресурс] / C. Bizer, T. Heath, T. Berners-Lee // International Journal on Semantic Web and Information Systems. – 2010. – Режим доступу до ресурсу: <http://tomheath.com/papers/bizer-heath-berners-lee-ijswis-linked-data.pdf>.
17. InfoXtract: A Customizable Intermediate Level Information Extraction Engine / R. K.Srihari, W. Li, C. Niu, T. Cornell. – Cambridge, England: Journal of Natural Language Engineering, 2008. – 723 с. – (Cambridge University Press).
18. Кондрашов В. Инвертований індекс [Электронний ресурс] / Володимир Кондрашов // up-promo.pro. – 2016. – Режим доступу до ресурсу: <https://up-promo.pro/seo-blog/vvedenie-v-poisk/invertirovannyij-indeks/>.

19. Manning C. D. Introduction to Information Retrieval / C. D. Manning, P. Raghavan, H. Schütze. – Cambridge, England: Cambridge University Press, 2009. – 581 с.
20. Егорова И.Н., Кадушкевич О.Н. Методика эффективного использования инструментов Google Analytics.- Науковий журнал “ScienceRise” №1/2(18)2016, ПП «Технологічний центр», с.40-44.
21. Егорова І.М. Проектування та розробка Web-документів: навч. посібник.- Харків: ХНУРЕ, 2018.- 264с. ISBN 978-617-669-261-4
22. Егорова И.Н., Кочура Л.А. Разработка обучающей игры «Web-технологии».- Вісник НТУ «ХП». Серія: Нові рішення в сучасних технологіях: зб. наук. пр. / Нац. техн. ун-т «Харків. політехн. ін-т». — Харків : НТУ «ХП», 2019. — № 2. — с.49-53
23. Егорова И.Н., Антипенко К.Д. О применении кривых Безье для улучшения CSS анимации.- Системи обробки інформації. Випуск 2(157). - Харків.-2019.- с.40-44
24. Егорова И.Н., Самокиш В.В. Об использовании спрайтовой анимации в веб-изданиях.- Системи обробки інформації. Випуск 3(154). - Харків.-2018.- с.100-104
25. Егорова И.Н., Гладкая А.А. Исследование возможностей резервного копирования веб-сайтов, созданных на основе WordPress.- Вісник Національного технічного університету «ХП». Збірник наукових праць. Серія: Нові рішення в сучасних технологіях. – Х.: НТУ«ХП». – 2017. –№ 23 (1245) – с.95-99
26. Егорова И.Н., Худолей А.Ю. Исследование возможностей компонентного подхода при разработке веб-сайтов.- Системи обробки інформації. Випуск 4(150). - Харків.-2017.- с.76-78
27. Егорова И.Н., Горелова Р.А. Разработка методики поисковой оптимизации веб-сайтов.- Системи обробки інформації. Випуск 4(150). - Харків.-2017.- с.73-75

28. Егорова И.Н., Филипенко О.В. Разработка методики создания графического интерфейса веб-сайтов.- Науковий журнал “ScienceRise” №1/2(18)2016, ПП «Технологічний центр», с.58-61