ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

Навчально-науковий інститут Інформаційних технологій

Кафедра Інженерії програмного забезпечення

Пояснювальна записка
до магістерської роботи
на ступень вищої освіти магістр

на тему: «**РОЗРОБКА МЕТОДИКИ АВТОМАТИЗОВАНОЇ ПОБУДОВИ ОПИТУВАЛЬНИКА НА ОСНОВІ КОНСПЕКТУ ЛЕКЦІЙ З ВИКОРИСТАННЯМ МЕТОДІВ TEXT MINING**»

Виконав: студент 6 курсу, групи ПДМ-61
спеціальності
121 Інженерія програмного забезпечення
(шифр і назва спеціальності)

Кононенко Ілля Віталійович
(прізвище та ініціали)

Керівник Садовенко В.С.
(прізвище та ініціали)

Рецензент
(прізвище та ініціали)

КИЇВ - 2022

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

Навчально-науковий інститут Інформаційних технологій

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти «Магістр»

Спеціальність 121 «Інженерія програмного забезпечення»

(шифр і назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Інженерії програмного
забезпечення
Негоденко О.В.
“ ” 2022 року

З А В Д А Н Н Я
НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТУ

Кононенкові Іллі Віталійовичу

1. Тема роботи: «Розробка методики автоматизованої побудови опитувальника на основі конспекту лекцій з використанням методів Text Mining»,
керівник роботи: Садовенко Володимир Сергійович, к.ф.-м.н., доцент кафедри Інженерії програмного забезпечення,

Затверджені наказом вищого навчального закладу від «12 жовтня» 2022 року № 122.

2. Строк подання студентом роботи 31.12.2022
3. Вихідні дані до роботи: методи Text Mining, Natural Language Processing, SQuAD, Науково-технічна література з питань, пов’язаних з програмним забезпеченням щодо інтелектуального аналізу тексту.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):

4.1 Теоретичні основи систем аналізу тексту з використанням методів Text Mining;
4.2 Розробка методики автоматизованої побудови опитувальника з використанням методів Text Mining;
4.3 Аналіз ефективності розробленого методу автоматизованої побудови опитувальника.

5 Перелік графічного матеріалу (презентація)
5.1 Актуальність дослідження;
5.2 Мета, об’єкт та предмет дослідження;
5.3 Text Mining;
5.4 Загальна структура Text Mining;
5.5 Приклади існуючих ІТ-рішень та їх моделей;
5.6 SQuAD як корпус автоматизованої побудови;
5.7 Особливості SQuAD;
5.8 QTA-204 як розроблена модель;
5.9 Формальна модель формування запитань;
5.10 Аналіз ефективності розробленого методу за перекладом;
5.11 Висновки.

5. Дата видачі завдання «14» жовтня 2022 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1.	Підбір науково-технічної літератури	20.10 – 12.11	
2.	Оцінка компонентів систем та інструментів і методів реалізації аналізу тексту	14.11 – 24.11	
3.	Розробка методики автоматизованої побудови опитувальника	24.11 – 14.12	
4.	Вступ, висновки, реферат	14.12 – 22.12	
5.	Розробка презентації	22.12 – 24.12	

Студент Кононенко І.В.
(підпис) (прізвище та ініціали)

Керівник роботи Садовенко В.С.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Текстова частина магістерської роботи 66 с., 10 рис., 36 джерел.

Об'єкт дослідження – використання методів Text Mining для автоматизованої побудови опитувальника.

Предмет дослідження – обмеження алгоритму побудови автоматизованого опитувального на основі методів Text Mining.

Мета дослідження – здійснити фундаментальний аналіз сучасного стану та особливостей розвитку методів Text Mining за допомогою, щоб розробити нову методику автоматизованої побудови опитувальника та модернізувати наявні методи використання алгоритмів Text Mining для майбутніх досліджень.

У роботі проведено аналіз існуючих програмних рішень, таких як IBM Intelligent Miner, ThemeScape, TextAnalyst та ін. та методів текстового аналізу.

Загальною проблемою цих рішень є те, що вони зазвичай неповні і не дають в повному обсязі провести аналіз тексту і згенерувати нові речення або питання і відповіді до них.

Особливістю розробленого методу є розбиття моделі на три рівні: рівень домену, рівень екстракторів, рівень генератора.

Це дозволяє зберігти усі дані, які вивчив штучний інтелект, витягти з них дані, і після цього згенерувати нові речення або питання з готовими відповідями до них.
Отже, розроблено та описано методику, завданням якого є покращення методів Text Mining.
Дана методика може цілком використовуватися як основа для створення програмного забезпечення з інтелектуального аналізу тексту.

Методи дослідження – сукупність загальнонаукових та спеціальних методів: методи наукової абстракції, аналізу, метод формалізації.

Галузь використання – сучасні системи інтелектуального аналізу тексту.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

TM – Text Mining. Інтелектуальний аналіз тексту.

DM – Data Mining. Інтелектуальний аналіз даних.

NaCTeM – National Center for Text Mining. Національний центр інтелектуального аналізу тексту.

NLP – Natural Language Processing. Обробка природної мови.

IF – Intermediate Form. Проміжна форма.

TFIDF – Term Frequency, Inverse Document Frequency) – Частота терміну, зворотна частота документа.

SQuAD – Stanford Question Answering Dataset. Стенфордський набір даних відповідей на питання.

CoVe – Contextualized Word Vectors. Контекстні вектори слів.

ELMo – Embedding from Language Models. Вбудовування з мовної моделі.

biLM – bidirectional Language Model. Двонаправлена мовна модель.

BERT – Bidirectional Encoder Representations from Transformers.

HMM – Hidden Markov Model. Прихована модель Маркова.

IO – Inside/Outside.

CRF – Conditional Random Field. Метод умовних випадкових полів.

BQG – Basic Question Generation. Базовий алгоритм для генерації питань.

SG – Sentence Generation. Генерація речень.

API – Application Programming Interface.

QTI – Question & Test Interoperability.

BPTT – Backpropagation Through Time. Зворотне розширення в часі.

LSTM – Long-short Term Memory. Модель довго-короткочаснаої пам’яті.

VES – Variable-Elasticity-of-Substitution. Модель змінної еластичності заміщення.

ROUGE-L – Recall-Oriented Understudy for Gisting Evaluation for Longest Common Subsequence based statistics.

BLEU – Bilingual Evaluation Understudy.

METEOR – Metric for Evaluation of Translation with Explicit Ordering.

ВСТУП

Актуальність дослідження: У зв’язку зі зростанням інтересу до методів інтелектуального аналізу текстових даних не лише представників наукової спільноти, а й широкої громадськості, стрімко зростає і кількість робіт, присвячених проблемам їх функціонування та подальшого розвитку.

Водночас глобальна економіка вільного та відкритого ринку переживає період незворотних трансформацій внаслідок формування абсолютно нового типу економічних і соціальних відносин. Усе це характеризується виробництвом знань, інтеграцією технологій та розвитком інформаційних децентралізованих мереж, які потребують впровадження нових форм аналізу чисельних потоків інформації, які кожної секунди передаються через них.

Сутність подібних мереж визначає основні переваги використання методів Text Mining, які безпосередньо розроблені з метою аналізу неструктурованих даних. В результаті різноманітні схеми та програми аналізу на основі подібних отримують все більшого поширення.

Однак, проблеми розвитку нових методів аналізу та витягу інформації з документів, пов'язані з надмірною складністю процесу аналізу, структурою природних мов та джерел інформації зумовлюють необхідність трансформації наявних підходів до кодування та аналізу текстових даних. В цьому контексті, актуальним стає дослідження методів Text Mining, їх класифікації, проблем функціонування та потенціалу до розвитку як інструменту витягу та трансформації даних, зокрема автоматизованої побудови опитувальника на основі документа.

Мета дослідження – здійснити фундаментальний аналіз сучасного стану та особливостей розвитку методів Text Mining за допомогою, щоб розробити нову методику автоматизованої побудови опитувальника та модернізувати наявні методи використання алгоритмів Text Mining для майбутніх досліджень.

Для досягнення поставленої мети необхідно розв’язати наступні завдання:

1. Встановити сутність поняття «Text Mining» і його основні етапи та класифікацію;
2. Висвітлити архітектуру та основні компоненти систем інтелектуального аналізу текстів;
3. Розглянути інструменти, програми та поширені методи реалізації аналізу тексту;
4. Надати характеристику SQuAD як навчальному набору даних;
5. Визначити методи та інструменти для розробки методу автоматизованої побудови опитувальника;
6. Надати опис розробленого методу автоматичної генерації питань за допомогою технології Text Mining;
7. Проаналізувати ефективність розробленого методу автоматизованої побудови опитувальника;
8. Визначити потенційні сценарії неправильного конфігурації;
9. Виокремити можливі сфери застосування розробленого методу та його потенціал до розвитку.

Об'єктом дослідження є використання методів Text Mining для автоматизованої побудови опитувальника.

Предметом дослідження є обмеження алгоритму побудови автоматизованого опитувального на основі методів Text Mining.

Методи дослідження: Для досягнення поставленої мети, розв’язання сформульованих завдань та отримання відповідних результатів використано сукупність загальнонаукових та спеціальних методів, які сприяли забезпеченню концептуальної єдності дослідження: методи наукової абстракції (виділення найбільш суттєвих особливостей розвитку Text Mining), аналізу (висвітлення переваг та недоліків інтелектуального аналізу текстових даних); метод формалізації (розробка методики автоматизованої побудови опитувального та пропозицій щодо модернізації у майбутньому).

Інформаційно-фактологічною базою дослідження виступили спеціальні монографії, посібники та спеціалізована література, а також окремі розділи з математичної статистики, математичних і статистичних методів аналізу тексту, загальної статистики та її методів аналізу; публікації вітчизняних та закордонних журналів, що містять елементи спостережень у сфері комп'ютерної лінгвістики; комп'ютерні опитування спеціалізованими компаніями, банками та національними комісіями з питань регулювання аналізу та регулювання потоків даних.

Зокрема, питанням розробки методів шифрування повідомлень на основі методів стенографії та процесів удосконалення подібних підходів ретельно досліджено зарубіжними та вітчизняними спеціалістами: Рао Б., Сінг С., Альтенберг Б., Патіл Н., Бейкон Д., Блейк К., Фінеган Е., Йоханссон С., Інтаояд В., Камйод К., Темді П., Карпіке Дж. Д., Грімальді П., Редігер Х., Кіран Ф., Гопал Х., Далві А., Ласт М., Данон Г., Рохер Д., Теноріо М., Вулф Дж. Х., Островська К., Єфіменко В., Мироненко С., Онищенко Є., Доценко І., Пентилюк М., Серажим К. та іншими.

Наукова новизна одержаних результатів полягає в тому, що на основні наукових здобутків визначено особливості застосування методів Text Mining не тільки з технічної точки зору (використання інтелектуального аналізу текстових даних для автоматизації побудови опитувальника), а й виокремлено цільові функції існування різноманітних методів аналізу, кодування та декодування тексту.

Практична значущість результатів дослідження: полягає в тому, що на основі проведених теоретичних досліджень було розроблено нову методику автоматизованої побудови опитувальника. Результати цього дослідження надалі дають змогу розробити нові інструменти та методи аналізу тексту з урахуванням результатів цього дослідження та міжнародного досвіду, перш за все – механізмів контролю та запобігання появи спотворень даних при проведенні аналізу.

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ СИСТЕМ АНАЛІЗУ ТЕКСТУ З ВИКОРИСТАННЯМ МЕТОДІВ TEXT MINING

1.1. Поняття і визначення Text Mining

Нещодавні технологічні досягнення призвели до появи нових типів спостережень і вимірювань, які раніше були недоступні, і які підживлювали тенденцію «великих даних». Поряд зі стандартними структурованими формами даних (що містять в основному числа), сучасні бази даних включають нові форми неструктурованих даних, що містять слова, зображення, звуки та відео, які потребують нових методів для використання та інтерпретації. Таким чином, сьогодні крім традиційних кількісних або якісних даних, знайомих статистикам, дані, які служать вхідними для алгоритмів вилучення інформації, можуть мати будь-яку форму, включаючи зображення, відео, аудіо або текст. Зокрема, джерела текстових даних для вилучення інформації можуть варіюватися від тексту у довільній формі до напівформатованого тексту (html, xml та інші) і включають ті джерела, які закодовані у форматах документів з відкритим кодом (напр., OpenDocument) та інших форматах (наприклад, Microsoft Word і Microsoft Powerpoint).

Відповідно для аналізу подібних джерел була розроблена концепція інтелектуального аналізу тексту яка також відома як статистична обробка тексту, виявлення знань у тексті, Text Mining або обробка природної мови, залежно від застосування та методології, яка використовується. Його також можна розглядати як розширення інтелектуального аналізу даних. Оскільки найприроднішою формою зберігання інформації є текст, вважається, що видобуток тексту має більший комерційний потенціал, ніж видобуток даних. Фактично, нещодавнє дослідження Вульфа Дж. показало, що 80% інформації компанії міститься в текстових документах [32].

Однак інтелектуальний аналіз тексту також є набагато складнішим завданням (ніж інтелектуальний аналіз даних), оскільки він передбачає роботу з текстовими даними, які за своєю суттю є неструктурованими та нечіткими.

Відповідно Text Mining (інтелектуальний аналіз текстових даних) стосується процесу вилучення цікавих і нетривіальних шаблонів або знань із текстових документів. При цьому текстовий документ може містити деякі структуровані поля, такі як назва, імена авторів, дата публікації, категорія тощо. Метою видобутку тексту є виявлення невідомої інформації, яка ще не відома та ще не була записана. Процес видобутку тексту починається зі збору документів із різних ресурсів. Інструмент інтелектуального аналізу тексту (спеціально розроблена програма) отримував би певний документ і попередньо обробляв його, перевіряючи формат і набори символів. Потім документ проходив би фазу аналізу тексту.

Text Mining (TM) можна використовувати для різних областей, починаючи від базових описів текстового вмісту через підрахунок слів і закінчуючи більш складними способами, такими як пошук зв’язків між авторами та оцінка вмісту сценаріїв (наприклад, автоматичне маркування есе). ТМ відноситься до процесу вилучення значущих числових індексів із тексту. Своїм походженням він зобов’язаний поєднанню різних суміжних галузей – інтелектуального аналізу даних (DM), штучного інтелекту, статистики, управління базами даних, бібліотекознавства та лінгвістики.

Його основна мета – обробити неструктуровану інформацію, що міститься в текстових даних, щоб зробити текст доступним для різних статистичних алгоритмів DM. Це може допомогти зробити текстові дані такими ж інформативними, як стандартні структуровані дані, і дозволить нам досліджувати зв’язки та закономірності, які інакше було б надзвичайно важко, якщо не неможливо, виявити. За допомогою ТМ інформацію, що міститься в тексті, можна класифікувати та кластеризувати з метою отримання таких результатів, як розподіл частоти слів,

розпізнавання образів і прогнозна аналітика, які можуть бути нелегко доступними за допомогою стандартних даних.

Можливість аналізу текстових даних визнана одним із головних елементів тенденції розвитку Big Data і провідним джерелом інформації для журналістики даних. Останніми роками глибше розуміння потенціалу ТМ змусило державні органи та приватні організації відігравати активну роль у розробці цієї технології. Так, наприклад, національний центр інтелектуального аналізу тексту (NaCTeM) був, мабуть, першим державним центром ТМ у світі, заснованим JISC Великобританії та керованим Манчестерським університетом [16].

NaCTeM було створено у 2004 році для надання послуг ТМ у відповідь на вимоги академічної спільноти Великобританії та забезпечення лідерства у її використанні в навчанні, викладанні, дослідженнях та адмініструванні. Потенціал ТМ також був визнаний в інших країнах світу. Наприклад, в Італії Cineca (консорціум, що складається з 54 італійських університетів, Міністерства освіти і дослідницьких центрів) використовує один із найпотужніших комп’ютерів у світі для проектування та розробки інформаційних систем і рішень ТМ для державного управління у сфері охорони здоров'я та бізнесу. ТМ також може бути стратегічним джерелом інформації, заснованої на фактах, яка може підтримувати процес прийняття рішень у різних сферах, від розробки політики до бізнесу. З цієї причини дослідники та практики з різних галузей використовують ТМ.

Загалом, основна мета ТМ полягає в тому, щоб перетворити текст на дані, які будуть придатними для аналізу. Щоб досягти цього, необхідно застосувати до текстових документів алгоритми штучного інтелекту, що потребують інтенсивних обчислень, і статистичні методи. Як зазначено в брифінгу JISC 2008 року, ТМ використовує широкий спектр завдань, які можна об’єднати в єдиний робочий процес, у якому можна виділити чотири різні етапи:

1. Пошук інформації.
2. Обробка природної мови (NLP)

3. Витяг інформації

4. Інтелектуальний аналіз даних.

Першим етапом ТМ є ідентифікація відповідних документів із великої колекції цифрових текстових документів. Використовувані інформаційно-пошукові системи спрямовані на ідентифікацію підмножини документів, які відповідають запиту користувача. Два приклади інформаційно-пошукових систем – це інструменти, що використовуються в бібліотеках для пошуку книг на певну тему, і веб-пошукові системи (наприклад, Google, Bing), призначені для пошуку інформації у Всесвітній павутині.

Після отримання підмножини текстових документів рядки символів мають бути оброблені для аналізу комп'ютерами. Комп'ютери повинні отримувати вхідні дані в певному форматі, щоб вони могли розуміти природні мови, як це розуміють люди.

Основна складність полягає в тому, що часто прихована структура природної мови є дуже неоднозначною. Хоча це може поставити під загрозу результат, розвиток NLP призвів до високого ступеня успіху в певних завданнях. Обробки природної мови дозволяє:

1. Класифікувати слова за граматичними категоріями (наприклад, іменники, дієслова);
2. Усунути неоднозначність значення слова серед багатьох значень, які воно може мати, виходячи зі змісту документа;
3. Синтаксичний розбір речення, тобто виконання граматичного аналізу, який дає нам змогу створити повне уявлення про граматичну структуру речення, а не лише визначити основні граматичні елементи в реченні.

На цьому етапі ТМ лінгвістичні дані про текст витягуються з документів, які все ще містять неструктуровану форму даних, і розмічаються до них.

Надалі для того, щоб бути видобутим, як і будь-який інший вид даних, неструктурований документ природною мовою має бути перетворений на дані в структурованій формі. Цей етап називається вилученням інформації, і це дані,

створені системами НЛП. Найпоширенішим завданням, яке виконується на цьому етапі, є ідентифікація конкретних термінів, які можуть складатися з одного або кількох слів, як у випадку науково-дослідних документів, що містять багато складних багатослівних термінів. Вилучення інформації також дозволяє пов’язувати імена та сутності (наприклад, людей та організацію, до якої вони належать) і більш складні факти, такі як зв’язки між подіями чи іменами.

Коли структурована база даних заповнюється інформацією, отриманою з анотованих документів, наданих алгоритмами NLP, дані нарешті готові до видобутку. У цьому контексті «видобуток» є синонімом «аналізу», оскільки метою є отримання корисної інформації з текстових даних для створення нових знань. Для цього, враховуючи, що дані тепер у структурованій формі, можна використовувати стандартні статистичні процедури та методи, застосовані до текстових даних, які тепер у структурованій формі.

Іноді цей етап повторюються багато разів, доки інформація не буде вилучена. Таким чином, вилучення знань охоплює пошук інформації, аналіз тексту, безпосереднє вилучення інформації, кластеризацію, категоризацію, візуалізацію, використання методів інтелектуального аналізу баз даних та методів машинного навчання. Загальну структуру інтелектуального аналізу тексту складається з двох компонентів:

1. Уточнення (вилучення) тексту, яке перетворює текстові документи вільної форми на проміжну форму;
2. Дистиляція (обробка) знань, яка виводить моделі або знання з проміжної форми.

При цьому проміжна форма цих двох етапів (IF) може бути напівструктурованою, як-от представлення концептуального графіку, або структурованою, як-от представлення реляційних даних. Проміжна форма також може бути документальною, де кожна сутність представляє документ, або

концептуальною, де кожна сутність уявляю собою об’єкт або концепцію інтересів у певній області.

Інтелектуальний аналіз IF на основі документів виводить шаблони та зв’язки між документами. Кластеризація, візуалізація та категоризація документів є прикладами майнінгу з IF на основі документів. Інтелектуальний аналіз IF на основі концепції виводить шаблон і зв’язок між об’єктами чи концепціями. Операції інтелектуального аналізу даних, такі як прогнозне моделювання та асоціативне відкриття, належать до цієї категорії. IF на основі документа може бути перетворений на IF на основі концепції шляхом перегрупування або вилучення відповідної інформації відповідно до об’єктів інтересів у певній області. Звідси випливає, що IF на основі документа зазвичай не залежить від домену, а IF на основі концепції є доменно-залежним.

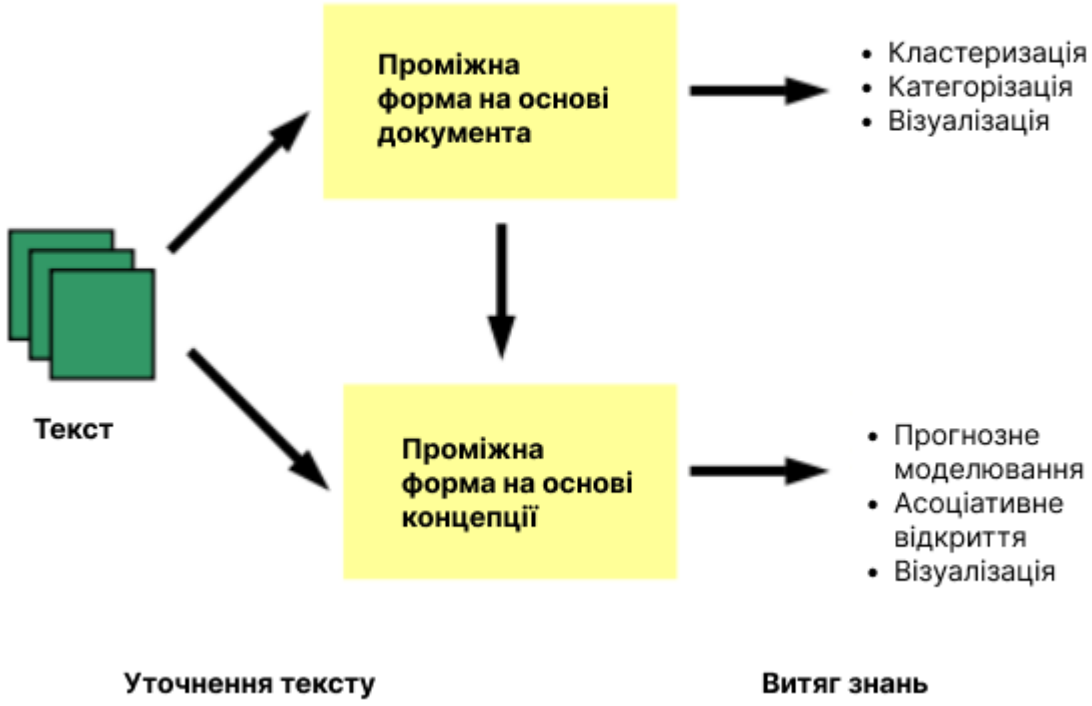


Рисунок 1.1. – Фреймворк інтелектуального аналізу тексту

Таким чином уточнення тексту перетворює неструктуровані текстові документи в проміжну форму (IF) (рис. 1.1). IF може базуватися на документі або на основі концепції. Дистиляція знань із документа IF виводить шаблони або знання в документах. IF на основі документа може бути спроектований на IF на основі концепції шляхом вилучення інформації про об’єкт, що стосується домену. Дистиляція знань із заснованого на концепції IF виводить шаблони або знання через об’єкти чи концепції.

Наприклад, для набору статей новин уточнення тексту спочатку перетворює кожен документ на IF на основі документа. Після цього можливо виконати дистиляцію знань на базі документів IF з метою організації статей відповідно до їх змісту для візуалізації та навігації. Для виявлення знань у певній області документальний IF новинних статей може бути спроектований на концептуальний IF залежно від вимог завдання. Наприклад, можна витягти інформацію, пов’язану з «компанією», з документа IF і сформувати базу даних компанії. Потім дистиляція знань може бути виконана в базі даних компанії (на базі компанії IF), щоб отримати знання, пов’язані з компанією.

Таким чином, першим кроком вилучення тексту є виділення функцій із колекцій документів, щоб можна було виконувати обчислення та застосовувати статистичні методології. Витягнуті функції повинні якимось чином відображати зміст документів. В ідеалі вони повинні фіксувати вміст таким чином, щоб документи, що обговорюють подібні теми, але з іншою термінологією, мали схожі характеристики.

Далі необхідно сформулювати спосіб вимірювання відстані між документами, де відстань будується, щоб вказати, наскільки вони схожі за змістом. Це не тільки дозволяє застосовувати методи класифікації та кластеризації, а також дає змогу сформулювати стратегії зменшення внутрішньої розмірності характеристик закодованого документа (мета зменшення розмірності полягає в тому, щоб забезпечити більш значущу геометризацію об’єктів).

Вище вже було наведено загальне визначення інтелектуального аналізу даних, однак надалі ми будемо використовувати більше точне трактування цього терміну, надане Девідом Хендом, Хейккі Маннілою та Падрейком Смітом у роботі “Принцип інтелектуального аналізу даних”. Вони визначили Text Mining як аналіз (часто великих) наборів даних спостережень з метою знайти непередбачені зв’язки та узагальнити дані новими способами, які є зрозумілими та корисними для власника даних.

Як було зазначено раніше, аналіз текстових даних стосується методологій аналізу даних, застосованих до текстових джерел. Документ – це послідовність слів і пунктуації, що відповідає граматичним правилам певної мови. Документ також – це будь-який відповідний сегмент тексту, який може мати будь-яку довжину. Приклади документів включають речення, абзаци, розділи, глави, книги, веб-сторінки, електронні листи тощо. При аналізі такоє використовується поняття термін, яка означає слово, пара слів або фразу. У цій роботі буде використовуватись термін і слово як синоніми. Набір (колекція) документів називається корпус, а лексикон – це сукупність усіх унікальних слів-термінів у корпусі.

1.2. Архітектура та оцінка компонентів систем аналізу тексту

Першою частиною вилучення ознак є попередня обробка лексикону. Зазвичай цей процес охоплює три методи:

1. Видалення стоп-слів;
2. Створення основи (корінь);
3. Зважування термінів.

Стоп-слова – це загальні слова, які не додають значущого змісту документу. Деякі приклади стоп-слів: і, але, або тощо. Стоп-слова можуть являти собою заздалегідь визначений список слів або вони можуть залежати від контексту чи

корпусу. Цей етап часто застосовується в області пошуку інформації, де метою є підвищення продуктивності системи та зменшення кількості унікальних слів.

Створення основи – це процес видалення суфіксів і префіксів, залишаючи корінь і основу слова. Наприклад, слова захищати, захищений, захищаючи та захист були б скорочені до кореня захист. Це має сенс, оскільки слова мають синонімічне значення. Однак деякі основоположні форми скорочують слова, що передають різні значення. Здебільшого, це не впливає на результати пошуку інформації, але може мати деякі небажані наслідки під час процесу класифікації та кластеризації. Визначення коренів і видалення стоп-слів загально зменшує розмір лексикону, таким чином заощадивши обчислювальні ресурси.

Один зі способів кодування тексту базується на періодично-частотних розрахунках (визначенні того як часто певні терміни зустрічаються у тексті). Однак терміни, які мають велику частоту, не обов’язково є більш важливими або мають вищу дискримінаційну здатність. Отже, формується необхідність зважити терміни відносно локального контексту, документа чи корпусу. Майже всі підходи зважування, які зараз використовуються, не базуються на теорії чи математиці, а є радше результатом вподобань авторів досліджень. Найбільш поширеною концепцією зважування терміну є зворотна частота документа, де частота терміну зважується відносно загальної кількості разів, коли термін з’являється в корпусі. Існує також розширення цього методу, яке називається зворотна частота документа (tfidf). Точне формулювання для tfidf наведено нижче у формулі 1.1.

$$w_{i,j} = tf_{i,j} * log(\frac{N}{df_i}) \tag{1.1}$$

де wi,j — вага терміну і в документі;

tfi,j – кількість появи терміна в документах;

j, N – загальна кількість документів;

а dfi — кількість документів, що містять термін і.

Розробка та розуміння впливу ваг термінів на методології інтелектуального аналізу текстових даних є ще однією сферою, де статистики можуть зробити свій внесок.

З огляду на те, чи було попередньо оброблено лексикон, вміст документів кодується. Одним із простих способів зробити це є використання векторного простору або підходу сумки слів. Перевага цього підходу полягає в тому, що він не потребує жодної обробки природної мови, наприклад ідентифікації частини мови чи перекладу мови. Корпус кодується у вигляді матриці термінів-документів X з n рядків і p стовпців, де n представляє кількість слів у лексиконі (повний або попередньо оброблений лексикон), а p – кількість документів. Таким чином, елемент xij матриці X містить кількість разів, коли i-й термін з'являється в j-му документі (частота терміну). Як ми зазначали раніше, це також може бути зважена частота. Кодування корпусу як матриці дозволяє використовувати надбання лінійної алгебри для аналізу колекції документів, як це робив Блейк у своїх дослідженнях (рис. 1.2).

D1: *Data Mining Techniques*: For Marketing, Sales, and *Customer Relationship Management*
D2: Principles of *Data Mining* (Adaptive Computation and *Machine Learning*)
D3: *Data Mining*: Practical *Machine Learning* Tools and *Techniques* with Java Implementations
D4: *Mastering Data Mining*: The Art and *Science* of *Customer Relationship Managemen*
D5: *Mastering Data* Modeling: A User-Driven Approach
D6: Investigative *Data Mining* for Security and *Criminal Detection*
D7: *Science* and *Criminal Detection*
D8: *Crime* and Human Nature: The Definitive Study of the Causes of *Crime*
D9: Statistics on *Crime* and *Criminals*: A Handbook of Primary *Data*

Term-Document Matrix:

	D1	D2	D3	D4	D5	D6	D7	D8	D9
crime (inal)	0	0	0	0	0	1	1	2	2
customer	1	0	0	1	0	0	0	0	0
data	1	1	1	1	1	1	0	0	1
detection	0	0	0	0	0	1	1	0	0
learning	0	1	1	0	0	0	0	0	0
machine	0	1	1	0	0	0	0	0	0
management	1	0	0	1	0	0	0	0	0
mastering	0	0	0	1	1	0	0	0	0
mining	1	1	1	1	0	1	0	0	0
relationship	1	0	0	1	0	0	0	0	0
science	0	0	0	1	0	0	1	0	0
techniques	1	0	1	0	0	0	0	0	0

Рис. 1.2. – Матриця termdocument (термін-документ)

Приклад невеликого корпуса із дев’яти книг про аналіз даних і розкриття злочинів; кожна назва є документом. Щоб заощадити місце, Блейк використовував в документі лише слова, виділені курсивом. ij-й елемент Матриця термін-документ показує, скільки разів і-те слово зустрічається в j-му документі.

Подібний підхід векторного простору у наступних дослідженнях було розширено, щоб включити порядок слів у значенні пар або трійок слів. Так, замість однієї матриці термін-документ можливо кодувати кожен документ як матрицю, яка називається матрицею близькості біграми внаслідок чого дослідники мали декілька матриць для роботи.

Під час підготовку до цього аналізу усі розділові знаки видаляються (наприклад, коми, крапки з комою, двокрапки, тире тощо), а всі розділові знаки в кінці речення

перетворюються на крапку. Період вважається словом у лексиконі. Таким чином, кожна матриця близькості біграми має n рядків і n стовпців (таблиця 1.1). Хоча подібні і формує простір з більшою вимірністю, ці матриці зазвичай дуже розріджені, тому ми можемо застосувати обчислювальні ефективні матричні методи для їх зберігання та аналізу.

Таблиця 1.1 – Приклад біграмної матриці близькості з одного речення

	.	влади	джерелом	є	знання	найдемократичнішим
.	0	0	0	0	0	0
влади	0	0	0		1	0
джерелом	0	0	0		0	1
є	0	0	0	0	0	0
знання	1	0	1	0	0	0
найдемократичнішим		1	0	0	0	0

Табл. 1.2 зображує біграмну матрицю близькості для речення: «Знання є найдемократичнішим джерелом влади» у якій лексикон був поставлений за алфавітом, а крапку поставили на початку.

Звичайно, існують і альтернативні підходи до аналізу слів. Одна нещодавно розроблена методологія кодує документ як єдиний рядок для підтримки аналізу в рамках машинної системи опорних векторів. Існують також загальнодоступні набори

інструментів обробки природної мови, які дозволяють виконувати теги частини мови та інші види операцій над їх корпусами.

Щоб розв’язати завдання інтелектуального аналізу текстових даних щодо кластеризації, класифікації та пошуку інформації, нам потрібно також визначити поняття відстані або подібності між документами. Найбільш часто використовуваною мірою в аналізі текстових даних і пошуку інформації є косинус кута між векторами, що представляють документи. Припустимо, що ми маємо два вектори документа ~a і ~q, тоді косинус кута між ними, θ, визначається на формулі 1.2:

$$cos(\theta) = \frac{\vec{a}^T \vec{q}}{||\vec{a}||_2 ||\vec{q}||_2} \quad (1.2)$$

де ||~a||2 – звичайна норма L2 вектора ~a.

Більші значення цього показника вказують на те, що документи розташовані близько один до одного, а менші значення вказують на те, що документи розташовані далі один від одного. Таким чином, ми можемо легко застосувати цю міру до матриць близькості біграми, перетворивши кожну матрицю на вектор, наприклад, просто наклавши кожен стовпець на інший. Однак, міра косинуса є мірою подібності, а не відстані. Тому, враховуючи, що зазвичай зручніше працювати з відстанями, краще перетворити подібності на відстані.

Для цього, по-перше, припустимо, що ми організували наші подібності в позитивно-визначену матрицю C (формула 1.3).

$$d_{ij} = \sqrt{c_{ii} - 2c_{ij} + c_{jj}} \quad (1.3)$$

де ij-й – це подібність i-го та j-го документів;

cii – перший документ;

cjj – другий документ.

Коли два документи однакові (cii = cjj), відстань дорівнює нулю.

Однак будь-якому корпусу притаманна висока розмірність характеристик документа, яка перешкоджає прямому застосуванню класифікації на основі ознак, такі стратегії, як лінійні/квадратичні класифікатори, змішані моделі та дерева класифікації, повинні поєднуватися зі стратегіями зменшення розмірності. Вони обговорюються в розділі 1.3. Проте є кілька підходів, які можна обговорити в цьому розділі.

Альтерберг досліджував застосування ієрархічної та модельної методології кластеризації разом із різними підходами дискримінантного аналізу для покращення якості аналізу англомовних текстів. Цікавим у цьому підході є те, що вони розробили метод дискримінантного аналізу, який можна застосовувати, коли кількість спостережень менша за розміри простору ознак [7].

Доценко у своєму огляду також зазначає роботи Ромеро, Маркес і Каррерас, які розробили нову модель навчання для прямої мережі, натхненну AdaBoost [3, с. 27].

Грімальді та ін. розробили схему категоризації гіпертексту, засновану на використанні орієнтованих ациклічних графових опорних векторних машин [18].

Треба також зазначити, що розробка схем класифікації текстових документів має ряд проблем. Однією з перших проблем є розв’язання проблем синонімії та полісемії. Іншим викликом є розробка ефективних схем кодування колекцій документів. Багато ще потрібно зробити на стику між простими підходами, заснованими на «мішку (сумці) слів», і більш складними схемами обробки природної мови. Іншим викликом є розробка схем класифікації, які можуть обробляти великі колекції документів. Останнім завданням, про яке я згадаю, є розробка схем класифікації джерел потокових документів. Ці джерела документів, наприклад дані новин, надходять безперервно. Потрібні методи аналізу такого типу даних.

1.3. Інструменти, програми та методи реалізації аналізу тексту

Перше інструменти ТМ з’явилося в середині 1980-х років. Сьогодні ТМ все частіше використовується в прикладних дослідженнях у різних сферах (таких як епідеміологія, економіка та освіта), а також у бізнес-цілях, особливо для отримання інформації про ринок і споживачів і для розробки нових продуктів. Техніки ТМ є спільними як для академічних досліджень, так і для бізнес-орієнтованої аналітики.

Деякі програми ТМ вимагають базової статистики, наприклад частоти. Підрахунок появи одного чи кількох слів у документі є найпоширенішим застосуванням ТМ, але для цього потрібні нові способи візуалізації такого роду даних. Наприклад, Wordle, безкоштовний інструмент, доступний онлайн, створює хмари тегів слів, що містяться в документі. Розмір кожного слова пропорційний його відносній частоті в документі (подібно до бульбашкової діаграми).

При цьому технологічний прогрес, який сприяв розвитку ТМ, не лише надихнув на нові візуалізації даних, але й стимулював збір нових «текстових баз», таких як Project Gutenberg і Google Books. Наприклад, оцифрування та архівування книг дозволяє нам обчислити частоту появи слова в книзі чи в усіх книгах, опублікованих у певному році, або візуалізувати випадки появи певних слів з часом. Для книг, доступних у Google Books, на рис. 1.3 подано приклад появи слів «інформація» та «новини» у книгах, виданих протягом останнього століття.

Рис. 1.3. – Пошук за словами «інформація» та «новини» в книгах Google у період з 2004 – 2022

Хоча слово «новини» (синє), здається, постійно використовувалося авторами протягом останнього століття, слово «інформація» зазнає помітного зростання: приблизно з такого ж рівня, як «новини» на початку 2000-х років, до шести разів більше, ніж «новини» у 2020 році.

Цей вид аналізу належить до нової галузі дослідження, відомої як «культуроміка». Наприклад, у нещодавньому дослідженні група дослідників зібрала вибірку з 7733 творів, отриманих із цифрової бібліотеки проекту Гутенберга, написаних 537 авторами після 1550 року [31].

Вони зосередилися на використанні 307 слів без вмісту (наприклад, прийменників, артиклів, сполучників і загальних іменників), стверджуючи, що ці слова забезпечують корисний стилістичний відбиток авторства та можуть використовуватися як метод порівняння авторських стилів. Для кожного автора був розрахований індекс схожості з кожним іншим автором. Цей індекс, заснований на повторюваності кожного слова без вмісту, розглянутого в дослідженні, використовувався для визначення часових тенденцій у використанні слів без вмісту. Їхній головний висновок полягав у тому, що автори, як правило, мають важливі стилістичні зв’язки з іншими авторами, ближчими за часом, але не обов’язково з безпосередніми сучасниками. Вони помітили, що для книг, виданих з інтервалом у три роки, індекс подібності дуже високий, але трохи нижчий, ніж у книжок, виданих з інтервалом у десять років. Для книг, опублікованих із тимчасовою відстанню понад десять років, індекс подібності зменшувався, доки не досягнув стабільного значення для книг, опублікованих із часовою відстанню 350 років.

Ще одне інноваційне дослідження, проведене Метью Джокерсом та Метью Келманом з Університету Небраска-Лінкольн, було зосереджено на порівнянні стилістичних і тематичних зв’язків між авторами вісімнадцятого та дев’ятнадцятого століть. Було оброблено величезну кількість текстових даних із використанням цифрових версій майже 3500 книг, щоб дослідити, як книги пов’язані одна з одною за такими критеріями, як частота слів, вибір слів і загальна тематика (рис. 1.4).

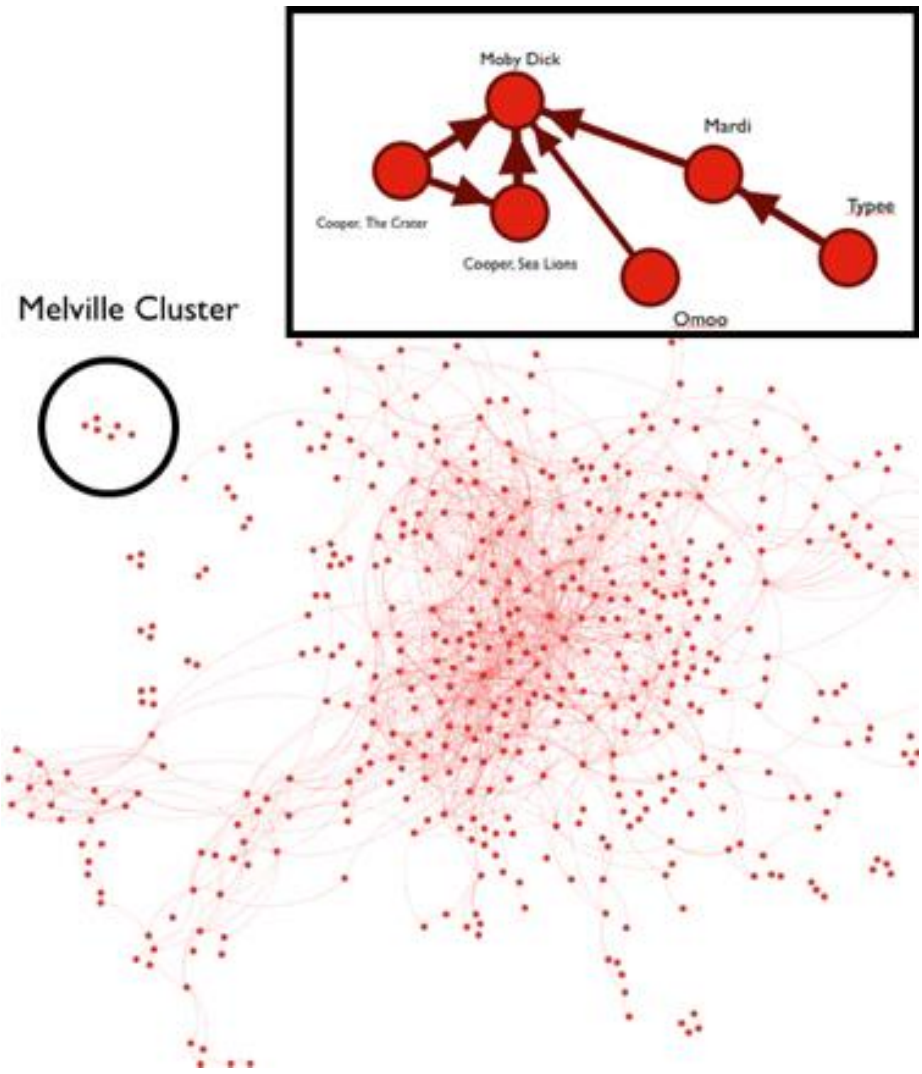


Рис. 1.4. – Графічний розподіл, який відображає зв’язки, ідеї та тенденції про літературний світ з кінця 1700-х до кінця 1900-х років

Кожна книга була скріплена унікальними атрибутами та відображена графічно. На рис. 1.4 показано книги, проаналізовані з кінця 1700-х до початку 1900-х років. Книги, намальовані ближче одна до одної, представляють тісний зв’язок з точки зору стилів і тем. Наприклад «Мобі Дік» Германа Мелвілла, опублікований в 1851 році, на графіку виглядає як відрив від більшості літературних творів того періоду, але все ще пов’язаний з кількома творами Джеймса Фенімора Купера («Морські леви», опубліковані в 1849 році, і «Кратер», опублікований у 1847).

У табл. 1.2 наведено ілюстративний список продуктів і додатків інтелектуального аналізу тексту на основі функцій уточнення тексту та дистиляції знань, а також прийнятої проміжної форми.

Таблиця 1.2 – Інструменти та програми для реалізації інтелектуального аналізу тексту на основі методів Text Mining

Компанія	**Інструмент**	**Функції уточнення тексту**	**Проміжна форма**	**Функції дистиляції знань**
Cartia	ThemeScape		На основі документів	Кластеризація, візуалізація
Canis	cMap		Гістограми слів на основі документів	Кластеризація, візуалізація
IBM/ Synthema	Technology Watch		На основі документів	Кластеризація, візуалізація
Inxight	VizControls		Гіперболічне дерево на основі документів	Візуалізація
Semio Corp	SemioMap		На основі концепції	Візуалізація

Продовження таблиці 1.2 – Інструменти та програми для реалізації інтелектуального аналізу тексту на основі методів Text Mining

Knowledge Discovery System	Concept Explorer	Інформаційний пошук	На основі концепції	
Inxight	LinguistX	Інформаційний пошук, аналіз тексту, реферування	На основі документів	
IBM	Intelligent Miner	Інформаційний пошук, реферування	На основі документів	Кластеризація, категоризація
TextWise	DR_LINK, CINDOR, CHESS	Інформаційний пошук, вилучення інформації	На основі концепції	
Cambio	Data Junction	Вилучення інформації	На основі концепції	
Megaputer	TextAnalyst	Інформаційний пошук, реферування	Документна семантична мережа	Класифікація

Перша група продуктів зосереджена на організації документів, візуалізації та навігації. До цієї категорії відноситься чимала кількість продуктів для аналізу тексту. Загальний підхід полягає в організації документів на основі їх подібності та

представленні груп або кластерів документів у певному графічному інтерфейсі. Наведений вище список аж ніяк не є вичерпним, але його достатньо, щоб проілюструвати різноманітність доступних схем представлення.

ThemeScape від Cartia – це корпоративна програма для відображення інформації, яка представляє кластери документів у ландшафтному представленні. Canis cMap – це інструмент кластеризації та візуалізації документів на основі самоорганізовних мап. IBM Technology Watch, розроблений спільно з Synthema в Італії, є програмою для аналізу тексту в науковій сфері. Він виконує кластеризацію документів і візуалізацію у вигляді карт для патентних баз даних і технічних публікацій. Inxight також пропонує інструмент візуалізації, відомий як VizControls, який виконує додаткову постобробку результатів пошуку шляхом кластеризації документів у групи та відображення на основі представлення гіперболічного дерева. SemioMap від Semio Corp використовує тривимірний графічний інтерфейс, який відображає зв’язки між концепціями в колекції документів. Зауважте, що SemioMap базується на концепції в тому сенсі, що вона досліджує зв’язки між концепціями, тоді як більшість інших інструментів візуалізації базуються на документах.

Друга група зосереджена на функціях аналізу тексту, зокрема пошуку інформації, виділенні інформації, категоризації та підсумовуванні. Хоча ми бачимо, що більшість систем інтелектуального аналізу тексту засновані на обробці природної мови, жоден із продуктів не має інтегрованих функцій інтелектуального аналізу даних для дистиляції знань між концепціями чи об’єктами.

Concept Explorer від Knowledge Discovery System – це інструмент візуального пошуку, який допомагає знаходити в Інтернеті точно відповідний вміст. Він «вивчає» зв’язки між словами та фразами автоматично зі зразків документів і візуально спрямовує вас до побудови пошуку. LinguistX від Inxight – це ще один інструмент для пошуку документів із можливостями аналізу тексту та резюмування. Intelligent Miner від IBM є, ймовірно, одним із найповніших продуктів для видобутку тексту. Він пропонує набір інструментів аналізу тексту, включаючи інструмент вилучення ознак,

набір інструментів кластеризації, інструмент підсумовування та інструмент категоризації. Також включено систему текстового пошуку IBM NetQuestion Solution і пакет веб-сканера IBM. TextWise, науково-дослідна компанія, що базується в Університеті Сіракуз, пропонує різні продукти інтелектуального аналізу тексту. DR-LINK – інформаційно-пошукова система, заснована на автоматичному розширенні поняття. CINDOR є його міжмовною версією. CHESS – це інструмент аналізу тексту та вилучення інформації. Також інструментом вилучення інформації є Cambio Data Junction, який витягує дані у формі реляційних атрибутів із тексту. A TextAnalyst Megaputer використовує семантичне мережеве представлення документів і виконує автоматичне індексування, призначення тем, абстрагування тексту та семантичний пошук.

Однак наведені вище інструменти семантичного аналізу є дорогими з обчислювальної точки зору і часто працюють із кількістю слів на секунду. Тому для полегшення візуалізації, кластеризації або класифікації застосовуються чисельні методи зменшення розмірності, що можуть видалити шум із даних і краще застосувати статистичні методи аналізу даних та інструменти наведені лише, щоб виявити тонкі зв’язки, які можуть існувати між документами.

Так, наприклад, ProjectIo – це інструмент, який призначений для зменшення кількості вимірів і який можна використати безпосередньо з матриці термін-документ. Інструмент побудований на принципі латентне семантичного індексування, яке є модернізованою теоремою сингулярного розкладання з лінійної алгебри.

Сингулярне розкладання дозволяє нам записати матрицю термін-документ як добуток трьох матриць (формула 1.4):

$$X = TSD \tag{1.4}$$

де, T – матриця лівих сингулярних векторів;

S – діагональна матриця сингулярних значень;

D — матриця правих сингулярних векторів.

Діагональні елементи S побудовані за домовленістю, щоб усі були позитивними та впорядкованими за зменшенням величини. Ліві сингулярні вектори T охоплюють стовпці X (простір документа), а праві сингулярні вектори D охоплюють рядки X (простір термінів).

Варто також зазначити, що кількість текстових запитів, які користувачі вводять у веб-пошукові системи, такі як Google і Yahoo, можна використовувати для прогнозного моделювання для прогнозування значень ряду цікавих показників. Дослідники в галузі епідеміології виявили, що пошукові запити на такі терміни, як «симптоми грипу» та «лікування грипу», були хорошим прогнозом кількості пацієнтів, яким у період 2004–2008 років потрібен був доступ до відділень невідкладної допомоги лікарні США протягом наступних двох тижнів.

Посилаючись на 2013 рік, повідомлялося, що ці веб-пошуки передбачали більш ніж удвічі більшу частку відвідувань лікаря з приводу грипоподібних захворювань, які були фактично зареєстровані. Ймовірно, це було спричинено зміною в алгоритмі пошуку Google [14].

Відповідно інтелектуальний аналіз текстових даних також можна застосувати до економічних показників для цілей, пов’язаних з бізнесом, і для аналізу думок клієнтів. Докази показали, що пошукові запити в Інтернеті «…можуть бути корисними провідними індикаторами для наступних покупок споживачів у ситуаціях, коли споживачі починають планувати покупки значно раніше, ніж прийняти рішення про покупку» [5]. Наприклад, дані пошукових систем, пов’язані із запитами щодо пошуку житла, виявилися більш точним прогнозом продажів будинків у наступному кварталі, ніж прогнози, надані економістами з нерухомості.

Однак, прогнозне моделювання на основі текстових даних виходить далеко за межі онлайн-світу. Одним із найвідоміших застосувань є розробка алгоритмів, які використовують текстові дані, що містяться в різних формах зв’язку (наприклад,

мобільні текстові повідомлення та електронні листи), для виявлення терористичних загроз і виявлення шахрайства у сфері охорони здоров'я та фінансових послуг.

Висновок до розділу 1

Отже, ключовою перевагою ТМ є можливість використання текстових записів у дуже великих масштабах. У цьому розділі ми коротко описали техніку ТМ і деякі її застосування. ТМ має різноманітні потенційні застосування в галузі освіти. Наприклад, у формувальному та підсумковому оцінюванні його можна використовувати для розуміння тенденцій у використанні лексики з часом, а також використання орфографії та пунктуації.

Розробки в області NLP дозволяють аналізувати мовну структуру величезної кількості текстових документів всього за кілька хвилин, а також поточні розробки в цій галузі можуть призвести до підвищення точності висновків. Наявність нових даних може привести до нових вимірювань і дослідницьких проектів для вирішення старих і нових дослідницьких питань. Однак, працюючи з дуже великими, насиченими та новими типами наборів даних, може бути непросто з'ясувати, на які питання дані можуть точно відповісти. Поставити правильне запитання зараз може бути важливішим, ніж будь-коли.

Використання великих текстових наборів даних без відповідного питання дослідження також може призвести до значної втрати ресурсів. Більш гетерогенні та глибокі дані могли б дозволити дослідникам перейти від методів, які дозволяють оцінювати середні співвідношення в популяції, до диференціальних ефектів для конкретних субпопуляцій, що представляють інтерес.

Отже, Text Mining – це сфера знань, що розширюється, і має потенціал для підтримки інноваційних сфер досліджень. Завдяки ретельному дослідницькому плану та відповідним методам ТМ може зробити вагомий внесок у майже будь-якій сфері життя людини.

РОЗДІЛ 2. РОЗРОБКА МЕТОДИКИ АВТОМАТИЗОВАНОЇ ПОБУДОВИ ОПИТУВАЛЬНИКА З ВИКОРИСТАННЯМ МЕТОДІВ TEXT MINING

2.1. Загальна інформація та аналіз обмежень SQuAD як корпусу

Історично великі, реалістичні набори даних грали критично важливу роль у процесі аналізу тексту. Відомі приклади включають ImageNet для розпізнавання об’єктів і Penn Treebank для синтаксичного аналізу. Однак більшість цих наборів даних мали один із двох недоліків:

1. Вони були високоякісні, але занадто малі для навчання сучасних моделей з інтенсивним використанням даних;
2. Або вони були великими, але напівсинтетичними та не мали тих самих характеристик, що й запитання для чіткого розуміння прочитаного.

Щоб задовольнити потребу у великому та високоякісному наборі даних про розуміння прочитаного, було розроблено Stanford Question Answering Dataset v1.0 (SQuAD), який є у вільному доступі на https://stanford-qa.com і складається із запитань, поставлених краудворкерами на набір статей Вікіпедії, де відповідь на кожне питання є фрагментом тексту, або проміжком, із відповідного уривка для читання. SQuAD містить 107 785 пар запитань-відповідей у 536 статтях, і це майже на два порядки більше, ніж попередні набори даних [21].

Таким чином, Stanford Question Answering Dataset (SQuAD) — це набір пар запитань і відповідей, які є серйозним викликом для моделей NLP. Як випливає з назви, SQuAD зосереджується на сфері відповідей на запитання. Він перевіряє здатність моделі прочитати уривок тексту, а потім відповісти на запитання про нього (ретроспективне розуміння прочитаного на SAT) (рис. 2.1).

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under **gravity**. The main forms of precipitation include drizzle, rain, sleet, snow, **graupel** and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals **within a cloud**. Short, intense periods of rain in scattered locations are called "showers".

What causes precipitation to fall?
gravity

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail?
graupel

Where do water droplets collide with ice crystals to form precipitation?
within a cloud

Рис. 2.1. – Зразок пар запитань і відповідей для уривка з набору даних SQuAD

Найважливішу частину створення набору даних – анотації – виконували працівники Mechanical Turk [27].

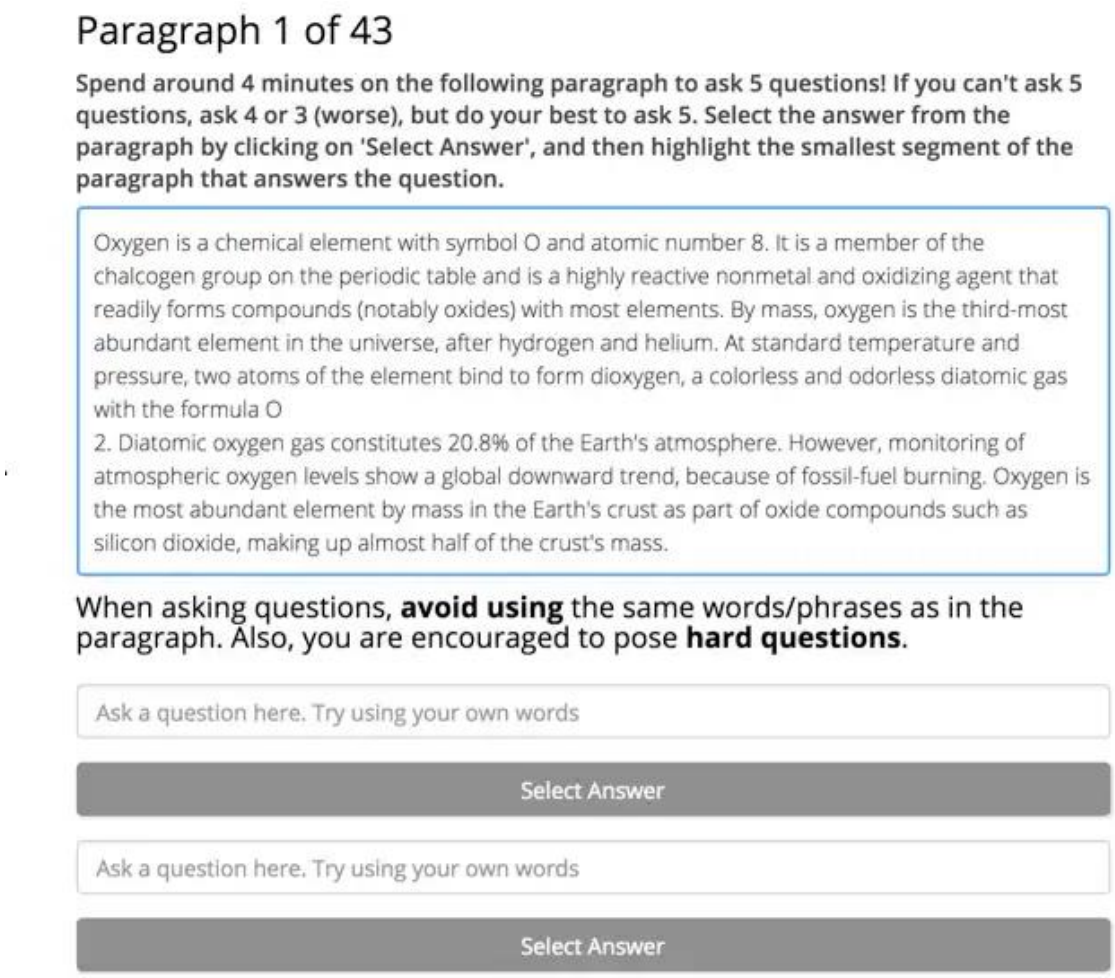


Рис. 2.2. – Процедура збору анотацій від працівників Mechanical Turk

До кожного обраного абзацу робітникам пропонувалося придумати і відповісти на 5 запитань за змістом абзацу. Їм було надано текстове поле для введення свого запитання, і вони могли виділяти відповіді в абзаці. Автори SQuAD подбали про те, щоб запитання, які придумували працівники, були їхніми словами, навіть відключивши функцію копіювання та вставки (рис. 2.2).

Основною перевагою SQuAD полягає у розумінні його властивостей. Для цього творці дослідили три напрямки:

1. Категорії відповідей. Кожна відповідь була розділена на одну з наступних категорій: «дата», «інше число», «особа», «місце», «інша сутність», «загальний іменник», «прикметник», «дієслово», «пункт» та «інше». Автори

виявили, що дати та числа становлять 19,8% відповідей, іменники становлять 32,6%, словосполучення іменників становлять 31,8%, а інші категорії становлять решту 15,8% (рис. 2.3).

2. Необхідне міркування. Автори SQuAD відібрали запитання з набору для розробки та вручну позначили запитання за різними категоріями міркувань, необхідних для відповіді на них. Наприклад, категорія «синтаксична варіація» означає, що питання, по суті, перефразовано та вимагає перестановки слів, щоб знайти відповідь.

2. Синтаксична дивергенція. Автори виміряли синтаксичну розбіжність між питанням і реченням, що містить відповідь, щоб виміряти складність запитання. По суті, вони створили метрику, яка оцінює кількість редагувань, необхідних для перетворення запитання у речення з відповіддю внаслідок чого було виявлено, що набір даних має різноманітні синтаксичні розбіжності.

Reasoning	Description	Example	Percentage
Lexical variation (synonymy)	Major correspondences between the question and the answer sentence are synonyms.	Q: What is the Rankine cycle sometimes **called**? Sentence: The Rankine cycle is sometimes **referred** to as a practical Carnot cycle.	33.3%
Lexical variation (world knowledge)	Major correspondences between the question and the answer sentence require world knowledge to resolve.	Q: Which **governing bodies** have veto power? Sen.: **The European Parliament and the Council of the European Union** have powers of amendment and veto during the legislative process.	9.1%
Syntactic variation	After the question is paraphrased into declarative form, its syntactic dependency structure does not match that of the answer sentence even after local modifications.	Q: What Shakespeare scholar **is currently on the faculty**? Sen.: **Current faculty include** the anthropologist Marshall Sahlins, ..., Shakespeare scholar David Bevington.	64.1%
Multiple sentence reasoning	There is anaphora, or higher-level fusion of multiple sentences is required.	Q: What collection does **the V&A Theatre & Performance galleries** hold? Sen.: **The V&A Theatre & Performance galleries** opened in March 2009. ... **They** hold the UK's biggest national collection of material about live performance.	13.6%
Ambiguous	We don't agree with the crowdworkers' answer, or the question does not have a unique answer.	Q: What is the main goal of criminal punishment? Sen.: **Achieving crime control via incapacitation and deterrence** is a major goal of criminal punishment.	6.1%

Рис. 2.3. – Приклади з набору розробок для кожної категорії міркувань, необхідних для відповіді на запитання

Порівнюючи SQuAD з іншими наборами даних, також можна виділити кілька особливостей:

1. SQUAD великий. Інші набори даних для розуміння прочитаного, такі як MCTest і Deep Read, занадто малі для підтримки інтенсивних і складних моделей. У MCTest лише 2640 питань, а в Deep Read лише 600 запитань. У SQuAD ці набори даних домінують із колосальними 100 000+ запитань

2. В інших наборах даних відповідей на запитання на основі документів, які зосереджені на вилученні відповіді, відповідь на задане питання міститься в кількох документах. Однак у SQuAD модель має доступ лише до одного проходу, що представляє набагато складніше завдання, оскільки пропустити відповідь вже не так пробачливо.

3. SQUAD вимагає аргументації. Популярним типом набору даних є набір даних cloze, який просить модель передбачити пропущене слово в уривку. Ці набори даних є великими, і вони представляють завдання, дещо схоже на SQuAD. Однак, ключове вдосконалення SQuAD у цьому аспекті полягає в тому, що його відповіді є більш складними і, отже, вимагають більш інтенсивного міркування. Таким чином, SQuAD є одним із найпопулярніших наборів даних із відповідями на запитання, який використовують у дослідженнях з метою оцінити модель NLP.

2.2. Визначення методів та засобів для побудови системи вилучення даних

Останнім часом методи інтелектуального аналізу текстових даних в основному базуються на машинному та глибокому навчанні, які відіграють вирішальну роль у вдосконаленні NLP. Для того, щоб перетворити проблеми NLP на проблему машинного навчання, символи, такі як текст, потрібно спочатку оцифрувати; тобто має бути отримано текстове представлення.

У 2016 році Bу та ін. популяризовали векторну модель слів, засновану статистичною мовою N-грам, яка стала піонером нейронної мережі як моделі мови [33].

З того часу було запропоновано багато представлень слів у низьковимірному просторі, які попередньо навчені на великому наборі текстового корпусу без міток. На відміну від високовимірного простору, ці представлення слів або вбудовування слів можна порівняти за семантичною віддаленістю та легко адаптувати до інших моделей.

Традиційно такі методи вбудовування слів, як Word2vec і Glove, були запропоновані для створення глобального представлення слів, яке враховує слово в усіх реченнях. В даний час все більше і більше робіт почали помічати різну семантику слова в різних контекстах. Наприклад, контекстні вектори слів (CoVe) фіксують контекстну інформацію за допомогою кодера в моделі машинного перекладу seq-to-seq, що привертає увагу; і вбудовування з мовної моделі (ELMO) витягує контекстно-залежні функції з двонаправленої мовної моделі (biLM) [2].

Згодом, завдяки пропозиції мережі Transformer, генеративного попереднього навчання Transformer (OpenAI-GPT) і великомасштабної моделі попереднього навчання на основі двонаправленого Transformer (BERT) попередньо навчені мовні моделі з Transformers для вилучення контекстного вбудовування слів показали кращу продуктивність, ніж будь-коли, у багатьох завданнях.

Однак, найбільшою популярністю у контексті побудови системи вилучення даних з текстів користуються саме методи контрольованого навчання, коли по корпусу розмічених даних (навчальній вибірці) будується модель (машинний класифікатор), яка застосовується до нових, нерозмічених текстів. Поряд із традиційними методами машинного навчання цього типу, такими як наївний баєсівський класифікатор, дерева рішень, метод опорних векторів, логістична регресія, все частіше застосовуються приховані моделі Маркова (Hidden Markov Models, HMM), метод умовних випадкових полів (condition random fields) та нейронні мережі. Дані, витягу яких необхідно навчитися, розмічені в текстах навчальної

вибірки певним чином, а саме записані їх лінгвістичні і структурні ознаки, часом також найближчий контекст. Для спрощення роботи експерта, який розмічає тексти, нерідко попередньо проводиться їх графематичний та морфологічний аналіз, рідше синтаксичний. Використовувана розмітка і ознаки, що враховуються, а також методи, що застосовуються для навчання, багато в чому залежать від виду інформації, що видобувається.

Для розмітки категорій іменованих сутностей запропоновано різні схеми. Найпростішим є схема IO (I – inside, O – outside), при застосуванні якої токени, що належать до імені (inside), розмічаються його категорією, а токени поза ім'ям (outside) розмічаються тегом O. Виходить, що якщо система навчається для розпізнавання імен персоналій (PERS), всі токени тексту діляться на два класи: які стосуються PERS і які стосуються O. У разі, якщо додатково враховуються географічні назви (LOC), то токени будуть ділитися вже на три класи: PERS, LOC або O.

Якщо система повинна навчитися розрізняти категорії PERS та LOC, то при навчанні всі токени тексту діляться на класи B-PERS, I-PERS, B-LOC, I-LOC та O, тобто класифікатор для кожного токена вчиться розпізнавати, чи є цей токен початком, чи продовженням іменованої сутності конкретної категорії (PERS чи LOC), і навіть випадок, коли токен взагалі є частиною найменування. Існують і більш складні схеми розмітки, що дозволяють розрізняти однослівні та багатослівні назви; для багатослівних назв передбачається розмітка як їх початку і продовження, так і закінчення. Для застосування машинного навчання дані після розмітки мають бути перетворені на набори ознак для кожного токена. Набір ознак зазвичай включає ознаки самого токена, а також ознаки, що ґрунтуються на знаннях, одержуваних із зовнішніх джерел, зокрема, зі словникових ресурсів. Причому ознаки вказуються як для значних (витягуваних) токенів, але й сусідніх, зазвичай беруться два токени зліва і справа.

До ознак токена зазвичай відносять:

1. Токен;

2. Вид токена: слово, розділовий знак, цифро-літерний комплекс тощо;
3. Довжину токена;
4. Позиція токена (чи є початком/кінцем пропозиції);
5. Тег токена, отриманий ним під час розмітки.

Для токенів-слів додатково враховується:

1. спосіб написання токена: тільки великими літерами, тільки малими, перша літера заголовна і т. п;
2. лема, частина мови, значення морфологічних ознак;
3. склад слова: коріння, суфікси та закінчення, типові для прізвищ, назв організацій та інших категорій сутностей.
4. Що стосується ознак, що отримуються зі словникових ресурсів, то вони, як правило, відповідають на питання, чи входить токен до певного словника. Використовувані словники можуть включати:
5. частотні імена, по батькові та прізвища, назви компаній, фірм та організацій, географічні назви;
6. слова, що є частинами найменувань, наприклад, типи організацій;
7. слова-маркери, за якими зазвичай розташовуються іменовані сутності певних категорій (місто, вулиця, річка тощо).

Використання під час отримання іменованих сутностей великої кількості словникових ресурсів є особливістю сучасних систем, заснованих на машинному навчанні (а також систем, заснованих на інженерному підході).

Таким чином, завдання вилучення іменованих сутностей сприймається як класичне завдання класифікації токенів кількох класів. При цьому для навчання та використання класифікаторів можуть застосовуватись різні стратегії.

Наприклад, можна навчити класифікатор одночасно розпізнавати сутності різних категорій, а можна побудувати окремі класифікатори кожної категорії і потім об'єднувати результати їх роботи. У той самий час, оскільки під час вирішення завдання вилучення сутностей активно використовується локальний контекст

класифікованого токена, дуже часто логічніше використовувати некласичні методи навчання (байесовський класифікатор, дерева рішень тощо), а приховані марківські моделі (HMM) та метод умовних випадкових полів (CRF), розглядаючи категорії іменованих сутностей як приховані стани, а токени – як спостережувані.

Сучасною тенденцією у розв’язанні завдання отримання іменованих сутностей є також застосування методів навчання без нагляду (методів кластеризації), що дозволяють автоматично кластеризувати слова за схожими контекстами їх вживання.

Важливо, що робота цих методів відбувається з нерозміченим текстовим корпусом, як вже згадувалось раніше, що дозволяє подолати обмеженість наявної текстової колекції. Зауважимо, що результати кластеризації іменованих сутностей можуть використовуватися також як додаткова ознака, що базується на знаннях, при застосуванні методів (часткового) нагляду.

В останні роки також з'явилися роботи, в яких застосовуються нейронні мережі та використовуються підходи на основі глибокого навчання (deep learning), наприклад, технологія Word2vec, але загалом вони не дали суттєвого приросту якості вилучення. Особливістю саме методів, що використовують нейронні мережі, є те, що вони дозволяють досягти якості, порівнянної з найкращими сучасними методами, але з мінімальним набором додаткової інформації: ознак токенів, словникових ресурсів та ін.

У завданнях розпізнавання відносин і фактів через складності розмітки даних використовуються дуже рідко. Найбільш типове використання методів з урахуванням часткового навчання – це так званий підхід distant supervision, у якому для навчання береться велика кількість прикладів сутностей (сотні та тисячі), пов'язаних певним ставленням або фактом. Джерелом цієї інформації може бути зовнішня база знань. Для формування навчальної вибірки у цьому контексті робиться досить грубе припущення, що обрані, що містять пов'язані певним ставленням сутності, є позитивними прикладами, а пропозиції, що містять сутності, але не пов'язані

цільовими відносинами, є негативними. Таким чином, автоматично готується навчальна вибірка пропозицій, до якої можна застосувати машинне навчання.

Ознаки, які застосовуються при вилученні відносин і фактів, що пов'язують іменовані сутності, в основному враховують контекст навколо сутностей:

1. Список лем слів, що стоять між сутностями, та їх частини мови;
2. Слова та їх частина мови зліва від лівої та праворуч від правої сутності;
3. Синтаксичний шлях між сутностями та його довжину;
4. Категорії іменованих сутностей.

Зазначимо також, що для коректної роботи системи вилучення, що базуються на машинному навчанні, розмічений корпус (навчальна вибірка) повинна мати досить великий обсяг, а також високу якість розмітки. Розмітка тексту є непростим і трудомістким процесом, який породжує досить високий відсоток помилок. Додатковою складністю може стати вибір відповідного методу навчання. Ще одне слабке місце підходу на основі машинного навчання пов'язане з тим, що результати роботи методів машинного навчання зазвичай погано зрозумілі, тому локалізувати і виправити помилки, що виникають практично неможливо.

При переході на інше завдання та предметну область системи, що використовують машинне навчання, стикаються з тими ж проблемами, що й системи, що ґрунтуються на правилах: систему необхідно налаштовувати заново. Залежно від використовуваного методу може знадобитися навчання системи на новому корпусі та/або коригування безлічі ознак, які він враховує. Проте вже розмічений корпус і створений набір ознак можна використовувати багаторазово, пробуючи на ньому різні методи і стратегії навчання, не залучаючи лінгвіста для кропіткої роботи з аналізу текстів предметної області та написання правил і шаблонів. Ця обставина пояснює широке використання машинного навчання в дослідницьких роботах із вилучення інформації в останні роки.

Ми перевірили досить різноманітний набір із 19 інструментів вилучення даних з текстів. Ці інструменти включали вільно доступні системи, розроблені в академічних

умовах, комерційні інструменти на основі API, які мали безкоштовний випробувальний період, і кілька алгоритмів, опублікованих у літературі з NLP. Хоча 19 включених інструментів аж ніяк не є вичерпним списком, вони включають інструменти з сотнями регулярних користувачів, ті, які були завантажені кілька тисяч разів, і ті, що з'являються в статтях, які цитуються сотні разів, і навіть використовують великі компанії.

Ці 19 інструментів можна загалом згрупувати у дві категорії:

1. Автономні комерційні інструменти;
2. Навчені робочі інструменти.

Окремі інструменти використовують моделі текстової аналітики, які можна застосувати безпосередньо до документів без міток відразу «з коробки». Ці інструменти включають пропозиції на основі API та ті, які можна завантажити як настільні програми. Ми включили в нашу оцінку 15 таких інструментів.

Готовий до використання характер цих інструментів, без потреби в розробці предметно-спеціальної моделі або навчанні, полегшує процес їх застосування. Однак відсутність специфіки домену також може бути шкідливою з точки зору продуктивності, оскільки моделі, що лежать в основі інструментів, можуть включати правила та/або припущення, які є помилковими або зайвими та неактуальними у контексті конкретного тестовому стенду.

Окремо було оцінено наступні комерційні інструменти: uClassify, ChatterBox, Sentiment140, Textalytics, Intridea, AiApplied, ViralHeat, Lymbix, SentimentAnalyzer, TextProcessing, Semantria, SentiStrength, MLAnalyzer і Repustate.

Більшість автономних інструментів, включених у дослідження, є комерційними пропозиціями, доступ до яких здійснюється безпосередньо через API постачальника або через сторонній ринок API, такий як MashApe. Двома винятками є Sentiment140 і SentiStrength, обидва з яких були розроблені в результаті опублікованих академічних досліджень. Ці інструменти використовують n-грами тегів слів і частин мови в поєднанні з класифікатором машинного навчання на основі максимальної ентропії.

Були також оцінені кілька інструментів верстака (Workbench) – це ті, які потребують контрольованої розробки моделі на основі навчання на визначеному навчальному наборі. Вони надають параметри для створення основи, токенізації, включення різних представлень функцій і параметрів для кількості функцій, які потрібно включити в моделі. Серед 5 інструментів верстака були LightSide, BPEF, EWGA, FRN і базовий запуск слова n-грам за допомогою розширення обробки тексту в RapidMiner.

Інструменти Workbench вимагають детального налаштування параметрів і перевірки в навчальному середовищі, але мають потенціал для включення точних завдань і предметних знань.

Наприклад, EWGA використовує ентропійно-зважений генетичний алгоритм для ефективного вибору ознак для класифікації термінів за допомогою моделі-обгортки, де продуктивність підмножини ознак використовується як значення функції відповідності в рамках генетичного алгоритму. FRN використовує мережу відношень ознак, що складається з двох ключових синтаксичних відношень n-грам: субсумпції та паралельних відношень. Ці відносини використовуються для ефективного вибору ознак із багатих просторів ознак, що охоплюють різноманітні типи n-грам. Скорочений набір функцій вводиться в класифікатор SVM. BPEF використовує структуру початкового параметричного ансамблю. Ансамбль, що охоплює десятки тисяч бінарних класифікаторів «один проти одного», створено з використанням різних комбінацій наборів даних, наборів функцій і машин.

Евристика пошуку на метарівні використовується для ідентифікації невеликої підмножини моделей, які зрештою зберігаються для класифікації. Базовий рівень n-gram складається з уніграм, біграм і триграм, вибраних за допомогою евристики отримання інформації та в поєднанні з класифікатором SVM, як це було зроблено в попередніх дослідженнях [6].

2.3. Автоматизована побудова опитувальника на основі розробленої системи вилучення знань

Сучасна освіта в багатьох предметних галузях сьогодні автоматизована. Однак, генерація запитань і тестових завдань є одним із напрямків, які не були добре автоматизовані, здебільшого внаслідок того, що генерація тестів є творчим процесом і часто вимагає досить багато часу. Якість ручних тестів може відрізнятися залежно від вимог, часу, витраченого на створення тестів, і кваліфікації особи, яка їх створює. Однак у сучасному світі для тестів часто не потрібні високі рівні складності. Але потрібна велика кількість тестів прийнятної якості та згенерованих у найкоротші терміни.

Тести бувають багатьох різновидів і можуть застосовуватися в ряді навчальних заходів, таких як контроль знань, залік, іспит тощо. Характер тестів для гуманітарних і технічних наук теж різний. Тому й підходи до їх створення мають відрізнятися. Існують різні інструменти автоматизованого тестування. Автоматизоване формування завдань передбачає формування тестових завдань на основі певних знань. Це родовище наразі мало досліджене. Крім того, попри те, що питання автоматизованої генерації тестів на даний момент є недостатньо вивченим, а коло поставлених проблем зовсім не торкається, існують дослідження, присвячені вирішенню задач автоматизованої генерації тестів. Одним із таких напрямків є підхід до автоматизованої побудови питань до тексту.

У роботі Капріке та інших пропонується створювати змістовні запитання до тексту, а потім використовувати лише ті, які найбільше відповідають критеріям викладача [18].

Іншим напрямком є автоматизована генерація тестів на основі моделі предметної області. Цей підхід передбачає побудову моделі, на основі якої система тестування генерує тести. Побудувати таку модель повинен фахівець, який обізнаний у відповідній предметній галузі. Підхід вимагає детальної декомпозиції домену,

побудови моделі, в якій визначаються закони отримання вихідних даних на основі вхідних даних і в яких заповнюються каталоги, за допомогою яких будуються тести. Такий підхід дає достатньо якісні тести, але потребує значного часу та зусиль, а також висококваліфікованих спеціалістів (для створення моделі предметної області), яких у деяких випадках бракує.

У цьому конкретному випадку під аналізом тексту розуміється пошук інформації в тексті за обраними шаблонами. Існує два основних типи шаблонів: шаблони для термінів і шаблони для класифікацій або властивостей. Шаблони термінів можуть включати такі шаблони, як «Термін – це визначення», «Термін означає визначення», «Визначення – це термін» та інші шаблони, що описують широко використовувані формальні записи визначень.

Шаблони класифікації подібні до шаблонів термінів. Вони будуються під широко поширені формати класифікацій та описів властивостей об’єкта чи поняття. З одного боку, цього аналізатора достатньо для функціонування системи на базовому рівні, з іншого боку, додаючи додаткові функції до аналізатора тексту, можна досягти кращих результатів на другому етапі – генерації тесту. Ця функція може включати семантичний аналіз тексту та побудову семантичних мереж. З семантичного аналізу нас цікавить, перш за все, визначення ключових слів у тексті.

Таким чином, в останні роки електронні тести стали переважаючим методом іспиту в українських університетах. Однак, незалежно від форми навчання, тестові питання складаються вручну одним або кількома викладачами, які працюють у групі. Це складний, трудомісткий процес. Тому для підтримки процедури тестування все більше уваги приділяється дослідженням, присвяченим розробці систем автоматичного та напівавтоматичного формування запитань. Нашою метою є розробка методики побудови опитувальника з можливістю (напівавтоматичного) створення тестових питань.

Ми представляємо QTA-204, керовану даними модель генерації запитань, яка слідує загальній структурі читача-генератора, але використовує кілька ключових нововведень, щоб забезпечити плавність і релевантність згенерованих питань.

Аналогічно запропонованій моделі контекстно-залежної агентної архітектури VES і моделі глобалізації запитів, ми намагаємося створити модель генерації запитань, яка буде використовувати методи Text Mining та NLP.

Наш підхід здебільшого універсальний і підходить до створення різних типів питань структурованого навчального контенту. Він складається з наступних трьох кроків:

1. Структуроване представлення знань, включених до навчальної програми. Знання поділяються на одиниці з відношеннями між ними.

2. Вилучення окремих компонентів, створення структури вмісту.

3. Створення запитань, використовуючи інформацію, отриману під час проведення кроку 2.

Відповідно до представленого підходу розробляється формальна модель, яка складається з трьох рівнів. Кожен рівень відповідає одному з кроків, запропонованих у підході:

1. Рівень 1 (домен): моделювання базових елементів і зв’язків між ними, структурування підтримуваного навчального контенту;

2. Рівень 2 (екстрактори): моделювання варіантів вилучення елементів зі структури вмісту;

3. Рівень 3 (генератор): моделювання генерації тестових запитань з використанням вилучених компонентів.

Формальна модель формування запитань виглядає наступним чином:

1. Рівень 1 (домен). Рівень 1 моделює знання предметної області як сховище (рис. 2.4).

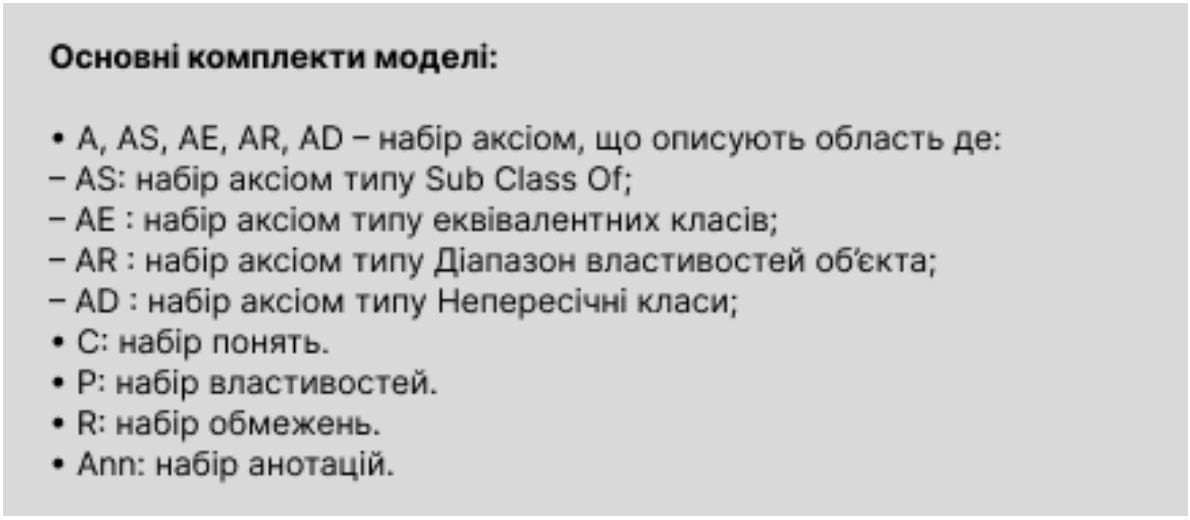


Рис. 2.4. – Визначення наборів і структур, пов'язаних із доменом (рівень домену)

Зміст навчання представлений як структура знань, що складається з понять, властивостей і аксіом. Концепції, що відображають сутності, які існують для конкретного домену, структуровані в ієрархію типів. Властивості представляють відношення між поняттями. Нарешті, аксіоми є специфічними для правил домену, які використовуються для визначення понять. Кожна аксіома складається з концепції, відомої як основа, і відповідного обмеження, яке допускається як уточнення базової концепції. Ми припускаємо, що описи правильні, тобто аксіоми завжди точні. Своєю чергою, обмеження складається з необов'язкової властивості та концепцій, відомих як обмежувальні концепції, тобто базова концепція може бути послідовно обмежена іншими концепціями стосовно даної властивості. Обмежувальні поняття можуть об'єднуватися або перетинатися відповідно до семантики, представленої обмеженням.

Обмеження, що містять властивість, ідентифікуються як анонімні концепції. Обмеження без властивості діють як заповнювачі базових понять в ієрархії. У структурі знань кілька типів аксіом використовуються для мети формування запитання.

Тип «Субкласи» описує відношення субсумпції між поняттями, включаючи анонімні поняття. Тип аксіоми «Еквівалентні класи» представляє еквівалентність між

поняттями, а тип аксіоми «Непересічні класи» характеризує протилежний зв’язок. Аксіоми непересічних класів описують відношення між поняттями, де вони не мають спільних рис. Аксіоми діапазону властивостей об’єкта надають інформацію про властивості та визначають їх діапазон. Якщо властивість використовується в обмеженні, концепція обмеження є концепцією для концепції діапазону властивості. Діапазон властивостей визначає набір понять, які можуть бути пов’язані цією властивістю.

Основним елементом моделі є набір анотацій. Анотації – це метазнання, які пов’язують додаткову інформацію з елементом у структурі. Анотація мітки є важливою для нашої моделі, оскільки вона забезпечує відповідну лексику елемента. Мітка властивостей ділиться на два типи – декларативні та питальні. Ці типи міток дозволяють граматично правильно використовувати властивість у реченні.

Рівень 2 (екстрактори). Рівень 2 моделі представляє екстрактори, які витягують відповідні знання зі структури знань у відповідній для генерації питання формі (рис. 2.5).

Набори:

- Axi – набір обраних аксіом за певним критерієм;
- K – набір критеріїв вибору аксіом;
- T {tI,tD} – набір типів речень, який складається з двох елементів, де:
  - tI – запитальний тип;
  - tD – декларативний тип.

Екстрактори:

- selectAxi: A → Axi – вибирає аксіоми, щоб ідентифікувати набір аксіом за певний критерій;
- extrBaseConcept: A → C – виділяє базове поняття з аксіоми;
- extrProperty: A → 2
- P – виділяє властивість з аксіоми;
- extrRestrConcept: A → C – витягує набір обмежувальних понять із обмеження в аксіомі;
- extrRange: P → C – витягує концепцію Діапазон властивості;
- extrAnnotations: A, P, C
  - Ann – витягує анотації сутності або аксіоми зі структури знань;
  - extrDisjointConcepts: C → C – виділяє непересічні класи.

Рис. 2.5. – Визначення наборів і структур, пов’язаних з екстракторами

На цьому рівні визначено набір типів речень. У моделі виділяються два типи речень – оповідальне та питальне. Необхідно розпізнати, який тип речення буде сформовано для запитання (генерація речень пояснюється на третьому рівні. Деякі екстрактори діють відповідно до типу речення, яке генерується. Множина Axi зберігає ряд аксіом, вибраних за певним критерієм (множина критеріїв позначається K). Критерій визначає тип аксіом і кількість понять в обмеженні. Він визначається відповідно до типу запитання, яке потрібно створити. Екстрактор selectAxi використовує критерії для вилучення необхідних аксіом зі знань. Існують екстрактори, які також витягують необхідні елементи зі структури вибраних аксіом.

Базовий концепт витягується extrBaseConcept, а властивість – extrProperty. Концепції обмеження малюються extrRestrConcept. Діапазон властивості витягується з відповідної аксіоми діапазону властивості об’єкта за допомогою extrRange. Екстрактор extrAnnotations використовується для вилучення анотацій елементів. Тип анотації, яку необхідно витягти, залежить від типу речення, яке має бути згенероване. Екстрактор extrDisjointConcepts малює непересічні поняття, які керуються аксіомами непересічних класів. Щоб проілюструвати роботу екстракторів, детально представлено функцію extrAnnotations. функція extrAnnotations (e, t) повертає анотацію даного елемента. Ця функція витягує всі анотації для заданого елемента та повертає одну, вибрану відповідно до заданого типу. Типи анотацій у базі знань створені таким чином, щоб вони відповідали типам речень.

Таким чином, екстратори формують мережу довготривалої короткочасної пам’яті (bi – LSTM), яка обробляє вхідну послідовність контекстних слів як у прямому, так і у зворотному напрямках (формула 2.1)

$$\overrightarrow{h_j} = \text{bi} - \text{LSTM}(\tilde{c_j}, \overrightarrow{h_{j-1}}) \tag{2.1}$$

де hj – це прихований стан, що відповідає прямому напрямку.

Потім при формуванні питального речення виділяється анотація типу «питальне». Спеціальна анотація потрібна головним чином для елементів властивостей, оскільки вони зазвичай представляють дієслова, які потребують різних словоформ відповідно до типу речення. Бувають такі випадки, як використання понять у реченні, коли певна форма слова не потрібна. Тоді типом анотації буде просто «мітка».

Рівень 3 (генератор). Генерація запитань послідовно виконується генератором у наступні три кроки, де використовуються дані, представлені на рис. 2.6:

- базовий алгоритм для генерації запитань (алгоритм BQG);
- генерація окремих речень, складання питання (алгоритм SG);

- генерація варіантів відповідей для типів запитань, що вимагають цієї функції.

Набори:

• Q – набір питань.
Крім того, існують два типи питань:
  • Q – набір класичних запитань;
  • QD – набір декларативних питань.
• Ans: набір відповідей.

Рис. 2.6. – Визначення наборів генераторів

Питальне речення є атомарними, оскільки містять лише одне питання. Оголошувальні питання складаються з одного або кількох оповідальних речень. Це загальні типи питань. Таким чином також визначається набір відповідей, оскільки деякі запитання можуть відповідати кільком варіантам відповіді, які користувач може вибрати.

Генератор запитань формує запитання слово за словом, від кроку часу t = 1 до t = L Q, де L Q – це довжина запитання, яку модель визначає за формулою 2.2. QTA-204 генерує питання шляхом ітеративної вибірки питальних слів qt ∈ V Q із розподілу ймовірностей (формула 2.2)

$$P(Q|C, A, \theta) = \prod_{t=1}^{L} P(q_t|C, A, q_{\tau}{}_{\tau=1}^{t-1}, \theta) \quad (2.2)$$

де θ позначає набір параметрів;

C – послідовність і-го вхідного контексту (наприклад, параграф із підручника);

L – кількість слів, у обраному фрагменті тексту;

A – довжина слів і пов’язаної послідовності відповідей;

P – патерн, який поєднує впорядковані пари з набору ключових слів і набору змінних відповідних типів.

a Q – вихідна послідовність запитань довжиною.

Цей дизайн моделі дозволяє генератору виводити запитання, які можуть зосереджуватися на певних частинах вхідного тексту. Наведені вище рівні моделі значно покращують плавність і релевантність згенерованих запитань. Спочатку зчитувач контексту обробляє кожне слово C у вхідному контексті та перетворює його на представлення hj ∈ R n фіксованого розміру. Далі генератор запитань генерує текст запитання, враховуючи всі репрезентації C. При цьому ключові слова відрізняються залежно від типу речення. Змінні витягуються з бази знань екстракторами. Результатом цієї моделі є цільове питання, яке складається зі згенерованих речень.

Спочатку дноспрямована мережа LSTM періодично відображає поточне (t-e) слово питання у вектор фіксованого розміру, який є t-м прихованим станом мережі:

$$s_t = \mathrm{LSTM}(y_t, s_{t-1}) \tag{2.3}$$

де, st ∈ R n – прихований стан, пов’язаний із t-м словом у запитанні.

Вектор c∗t∈R n є контекстним вектором, який є зваженою сумою вхідних прихованих станів:

$$c_{*t} = H_a \tag{2.4}$$

де C*t – це вектор ваги уваги, який розраховується як:

$$a_t = \mathrm{softmax}(H^T W_{hst}) \tag{2.5}$$

де Whst є частиною параметрів моделі.

Залежно від типу речення, наборам ключових слів і змінним прийнятних типів призначаються відповідні значення. Це фактичні значення, які будуть використані для заповнення шаблону. Питання будуть запитувати базове поняття з аксіоми. Таким чином обмежувальні поняття будуть відповідями. Конструктор створює остаточну форму речення, використовуючи заданий шаблон. У разі порожнього набору ключових слів речення формується елементами зі змінних прийнятних типів. Змінні розміщуються в реченні в тому ж порядку, що й у наборі. Порядок такий же, як і в оригінальній аксіомі, і він підходить для декларативних речень. Якщо обидві множини містять елементи, то впорядковані пари з множин розташовуються послідовно в реченні шляхом об'єднання. За своїм типом запитання можуть відповідати варіантам відповідей. Для цього розроблено функцію відповідей. Алгоритм відповідей вибирає серед понять відповідні варіанти відповідей.

Варіанти відповідей використовуються по-різному, залежно від типу питання. Коли вони відповідають на класичне запитання, вони представлені як окрема частина речення, тому користувач може вибрати одну або кілька правильних відповідей. Якщо запитання має декларативний характер (складається з одного чи кількох речень із декларативним змістом), варіанти відповідей подаються як частини речень. Деякі поняття в реченнях доречно замінити варіантами відповідей. Це дає можливість дати завдання для визначення правильності речень або пошуку помилок. Функція вибирає ряд понять із набору обмежувальних понять в аксіомі для генерації. Це правильні відповіді, які відповідають цьому реченню.

Висновок до розділу 2

Незважаючи на ці проблеми, останніми роками машинне навчання стає все більш популярним у дослідженнях вилучення інформації, про що свідчить широке використання різноманітного набору з 19 інструментів вилучення тексту, включаючи як безкоштовні академічні системи, так і комерційні інструменти на основі API з безкоштовними пробними періодами, а також алгоритми, опубліковані в літературі з

обробки природної мови, які використовуються великими компаніями та чисельно цитуються у різноманітних наукових статтях.

Використання машинного навчання при розробці нової моделі потребує великої, якісної навчальної вибірки. Для цього ми обрали и використали Стенфордський набір даних відповідей на запитання (SQuAD), який включає понад 100 000 екземплярів даних, які складаються з абзацу, взятого зі статті у Вікіпедії, відповіді та запитання, створеного людиною на основі абзацу і відповісти.

Основною перевагою нашої моделі буде той факт, що під час переходу до іншої задачі чи предметної області наш метод, заснований на машинному навчанні, можна буде переконфігурувати легше, ніж системи, засновані на правилах, оскільки вже позначений корпус і набір функцій можна повторно використовувати та перевіряти за допомогою різних методів і стратегій навчання, не вимагаючи залучення лінгвістів для ручного аналізу текстів і написання правил і шаблонів.

Таким чином, наша модель є результатом розробки автоматичного середовища для генерації тестів, що містять ряд питань. Модель буде розширено, щоб вона визначала певні методи для створення різних типів питань. Також варто зазначити, що для визначення типів запитань у майбутньому можна буде використовувати вже готові стандарти, наприклад QTI (IMS Question & Test Interoperability). Подібні специфікації пропонують добре задокументований формат для розробки частин, необхідних для навчання та тестування систем електронного навчання. Використання такого стандарту для визначення формату тестів і питань дає більше можливостей для роботи з різними елементами, такими як зберігання та обмін інформацією.

РОЗДІЛ 3. АНАЛІЗ ЕФЕКТИВНОСТІ РОЗРОБЛЕНОГО МЕТОДУ АВТОМАТИЗОВАНОЇ ПОБУДОВИ ОПИТУВАЛЬНИКА

3.1. Аналіз ефективності використання розробленого методу

Щоб оцінити продуктивність моделі генерації тексту, можна використовувати різноманітні показники. Деякі загальні показники для оцінки моделей подібній нашій включають показники заплутаності, бали BLEU, METEOR, ROUGE-L та традиційну рецензію.

Однак, оскільки навчальний контент зазвичай не має певного контексту, пов’язаного з кожним питанням, ми можемо лише кількісно оцінити нашу модель, порівнюючи її з базовими показниками за допомогою загальнодоступних наборів даних загального призначення. Тому навчати нашу модель можливо на SQuAD, Стенфордському наборі даних відповідей на питання (як вже було зазначено у розділі 2).

SQuAD містить понад 100 тисяч екземплярів даних, кожен з яких складається з короткого абзацу, взятого зі статті у Вікіпедії, відповіді, яка є фрагментом тексту з абзацу, і створеного людиною запитання на основі абзацу та відповіді. Ми розглядаємо абзац як вхідний контекст для моделі, а питання – як вихідний, таким чином фактично перетворюючи SQuAD на навчальний набір даних для створення запитань. Набір даних явно надає нам індекси першого та останнього слів у відповіді. Ця інформація спрощує кодування відповіді у відповідних векторах слів контексту. Ми скорочуємо кожен абзац лише до одного речення, яке містить відповідь, і використовуємо це речення як контекст під час навчання, як це вже було зроблено у дослідженнях Доценко [3].

Набір даних SQuAD складається з навчального набору, набору перевірки та тестового набору. Всі вони мають однаковий формат. Оскільки тестові дані приховані і до них неможливо отримати доступ, ми розділили набір перевірки на дві половини:

одну половину використовуємо для перевірки, а іншу – для тестового набору. Під час навчання ми прагнемо мінімізувати різницю між згенерованим запитанням і справжнім запитанням у навчальному наборі, тому ми кількісно визначаємо цю різницю, використовуючи від'ємну логарифму правдоподібності (форм. 3.1)

$$L(\theta) = -\log P(Q|C, A, \theta) = -\sum_{t=1}^{L} \log P(q_t|C, A, q_{i\tau}{}_{\tau=0}^{t-1}, \theta). \quad (3.1)$$

де, C – послідовність і-го вхідного контексту (наприклад, параграф із підручника);

L – кількість слів, в обраному фрагменті тексту;

A – довжина слів і пов'язаної послідовності відповідей.

Оскільки ця функція втрат є диференційованою скрізь, ми використовуємо стандартне зворотне поширення в часі (BPTT) з міні-пакетним алгоритмом стохастичного градієнтного спуску для вивчення параметрів моделі. Щоб створити найкраще запитання, ми використовуємо пошук за променем, щоб вибрати 25 можливих кандидатів на виведення питальних речень. Потім ми вибираємо питання з найнижчим від'ємним логарифмом у якості кінцевого запитання.

Ми порівнюємо QTA-204 із такими базовими рівнями: Overgenerate & Rank, система на основі правил, яка досягає порівнянної продуктивності з моделями на основі нейронних мереж; базова модель LSTM.

Автоматичне оцінювання моделей генерації запитань є складним завданням, оскільки немає метрик, розроблених спеціально для вимірювання якості запитань. Тому ми використовуємо BLEU і METEOR з машинного перекладу, а також ROUGE-L. Ці показники обчислюються шляхом порівняння запитання, створеного машиною, із запитанням, створеним людиною, з тих самих вхідних даних.

Таблиця 3.1 – Порівняння між QTA-204, LSTM та Overgenerate & Rank

Модель	**Метрика**		
	BLEU	METEOR	ROUGE-L
Overgenerate & Rank	0.1120	0.1702	0.2792
LSTM	0.0231	0.0796	0.2703
QTA-204	0.1386	0.1838	0.4437

Усі оцінки метрик приймають значення в [0,1] (табл. 3.1); вищі значення вказують на питання вищої якості. Ці показники слугують початковим, доступним і широкомасштабним порівнянням між нашою моделлю та декількома іншими базовими рівнями та можуть виявити розуміння плавності та актуальності запитань, створених кожною моделлю. Ми бачимо, що QTA-204 перевершує за всіма базовими показниками SQuAD, іноді значно.

Варто також зазначити, що продуктивність QTA-204 покращується, коли стає доступним більше навчальних даних. Ми застосовуємо ту саму конфігурації навчаючи QTA-204 на 10%, 50%, і 100% навчального набору SQuAD (табл. 3.2).

Таблиця 3.2 – Приклади питань, згенерованих моделями QTA-204, LSTM та Overgenerate & Rank

Модель	**Метрика**		
	10%	50%	100%

Продовження таблиці 3.2 – Приклади питань, згенерованих моделями QTA-204, LSTM та Overgenerate & Rank

Overgenerate & Rank	When Albert Einstein was nominated for a Nobel Prize in Physics?	When Albert Einstein was nominated for a Nobel Prize in Physics?	When Albert Einstein was nominated for a Nobel Prize in Physics?
LSTM	What is nominated for a Nobel Prize?	In what year Albert Einstein was nominated for a Nobel Prize?	In what year Albert Einstein was nominated for a Nobel Prize?
QTA-204	When was Einstein nominated for a Nobel Prize?	When was Einstein nominated for a Nobel Prize?	When was Einstein nominated for a Nobel Prize?

Ці результати показують, що продуктивність моделей на основі нейронної мережі підвищується з більшою кількістю навчальних даних, тоді як продуктивність базової системи на основі правил – ні. Ми також спостерігаємо, що QTA-204 перевершує базовий рівень LSTM, навіть коли обсяг навчальних даних дуже низький.

Як бачимо на табл. 3.2 модель Overgenerate & Rank на основі правил генерує одне й те саме запитання, незалежно від кількості навчальних даних. У той час як обидві моделі на основі нейронних мереж генерують запитання вищої якості з більшою кількістю навчальних даних. Крім того, QTA-204 швидко вчиться зосереджуватися на відповідній інформації та може створювати запитання навіть з неповними навчальними даними.

Таким чином, QTA-204 успішно фіксує тонку різницю між відповідною інформацією і генерує відповідне запитання для кожної відповіді. Крім того, запитання, згенеровані QTA-204, є більш зрозумілими та релевантними для вхідних контекстів і відповідей, ніж ті, які генеруються іншими проаналізованими моделями.

Перенесення моделі QTA-204, навченої на наборі даних загального призначення SQuAD, до конкретних предметних областей може призвести до певних проблем. Зокрема, якщо ми візьму нашу навчену модель і застосовуємо її до навчального контенту без будь-яких додаткових налаштувань.

Наприклад, візьмемо ми можемо взяти три підручники з бази даних OpenStax [25]. Для кожного підручника ми генеруємо одне запитання на пару контекстних відповідей, використовуючи три різні моделі: QTA-204, Overgenerate & Rank і LSTM.

Далі нам потрібно залучити рецензентів. В цьому дослідженні рецензентом виступав безпосередньо автор та чотири запрошені особи. Щоб уникнути можливих упереджень, випадковим чином перемішуємо порядок представлення трьох запитань для кожного вхідного контексту.

Після цього просимо надати оцінку (так чи ні) для кожного з трьох запитань за двома показниками:

- чи є запитання вільним (тобто зв’язним і граматично правильним);
- чи відповідає питання вхідним даним пара контекст-відповідь.

В якості остаточного показника (під назвою «уподобання») просимо вибрати питання, які могли бути створені справжньою людиною і дозволяємо рецензенту вибрати більше одного запитання або жодного з них. Цей показник відображає суб’єктивні оцінки щодо того, наскільки «людськими» є запитання.

Далі ми розраховуємо кількість питань, у яких більшість оцінювачів дали позитивну оцінку для кожної з трьох моделей генерації запитань.У всіх випадках QTA-204 (часто значно) перевершує дві інші моделі за всіма трьома показниками оцінювання. Ми оцінюємо статистичну значущість цих результатів за допомогою біноміального тесту та знаходимо, що ступінь, до якого QTA-204 перевершує два

базові показники, є статистично значущою далеко за межами рівня p = 0,05 для всіх показників оцінки, за винятком вільності та відповідності для підручника з історії. (p = 0,11 і p = 0,5 відповідно).

Таким чином, ми робимо висновок, що QTA-204 генерує запитання, які є вільними, релевантними та «схожими на людину» частіше, ніж існуючі моделі. Ці результати означають, що питання, створені QTA-204, краще застосовні в реальних освітніх умовах, ніж ті, що створені іншими базовими лініями.

3.2. Визначення контексту використання розробленого методу

Автоматична генерація побудови опитувальника на основі конспекту значно скорочує час викладачів на побудову вправ та тестових завдань. У цьому контексті було б цікаво оцінити час, необхідний для створення такої ж кількості питань без QTA-204, щоб наскільки можуть вчителі-експерти заощадити час та власні зусилля за допомогою системи. Однак, QTA-204 має і певні недоліки.

По-перше, наш метод здатний генерувати лише фактичні запитання. Хоча фактичні запитання є цінними для навчання (як ми вже зазначали у розділі 2.3), це обмежує глибину питань. Крім того, існує кілька сценаріїв неправильної конфігурації, коли QTA-204 може бути непридатною для створення тестових запитань:

1. Відсутність відповідних даних: хоч наша модель і продемонструвала високу ефективність (порівняно з двома іншими моделями) при розробці тестів базуючись на невеликій базі даних. Здебільшого наша модель покладається саме на великий і різноманітний набір даних, щоб вивчати шаблони та генерувати зв’язний і граматично правильний текст. Якщо система не має доступу до відповідного набору даних, вона може не мати змоги згенерувати відповідні та точні тестові запитання.

Наприклад, наявна конфігурація QTA-204 навчена лише на наборі даних SQuAD та не була піддана жодним текстам Шекспіра чи літературній термінології загалом.

Відповідно, моделі бракує необхідних знань і досвіду для створення якісних тестових запитань із цієї теми.

Як результат, коли моделі дали завдання сформувати питання пов’язані з цією темою, вона зробила багато помилок, а саме:

1. «Who wrote the play 'Shakespear'?» (питання некоректне, оскільки Шекспір – це людина, а не п’єса);

2. «What is the capital of England?» (нерелевантне питання, оскільки тема шекспірівська література, а не географія)

3. «What is the main character's name in the play?» (некоректне запитання, оскільки не було вказано назви п’єси);

4. «What is the meaning of the word 'thou'?» (некоректне питання, оскільки «ти» — це архаїчна форма «ти», а не конкретне слово з унікальним значенням)

Це лише кілька прикладів неякісних тестових запитань, які QTA-204 може генерувати за сценарію несправної конфігурації. Саме тому важливо ретельно оцінити якість і релевантність згенерованих запитань, перш ніж використовувати їх у будь-якому тесті.

Таким чином, попри те, що QTA-204 генерує значно кращі запитання, ніж проаналізовані у розділі 3.1 моделі, наша модель не гарантує завжди створює хороші запитання, оскільки не існує ефективних метрик оцінки запитань, які могли б автоматично відфільтрувати погані запитання. Відсутність таких інструментів означає, що QTA-204 ще не готова до широкомасштабного автоматизованого розгортання в реальних освітніх умовах, оскільки експерти-люди все ще повинні переглядати згенеровані запитання перед передачею їх учням.

Недивний, але важливий факт дослідження полягає в тому, що більші моделі, як правило, працюють краще. Однак завжди будуть сценарії, де використання меншої моделі подібної QTA-204 є корисним, наприклад. У зв’язку з цим, одним з корисних способів використання розробленої моделі є можливість досягти гарної продуктивності у завданнях з низькими ресурсами. Завдання з низьким ресурсом

часто виникають (за визначенням) у налаштуваннях, де не вистачає ресурсів, щоб позначити більше даних.

У зв’язку з цим ми виступаємо за дослідження методів, які забезпечують більшу продуктивність за допомогою дешевших моделей, щоб можна було застосувати трансферне навчання там, де воно матиме найбільший ефект, наприклад:

1. Освітні цілі: вчителі та професори можуть використовувати QTA-204 для створення нових унікальних екзаменаційних запитань або вправ для есе. Це може допомогти запобігти шахрайству та ускладнити студентам пошук попередньо розв’язаних опитувальників, тестів, завдань тощо.

2. Дослідницькі цілі: спеціалісти в різних галузях можуть використовувати інструмент формування запитань, щоб сформувати нові ідеї для дослідницьких проектів або створити запитання для опитувань чи фокус-груп. Це може допомогти дослідникам визначити нові сфери дослідження та створити запитання, які мають відношення та мають значення для теми дослідження.

3. Практика письма: студенти або письменники, які хочуть покращити свої навички написання есе, можуть використовувати QTA-204 для створення запитань, щоб знайти ідеї для практичних есе. За правильною конфігурацією QTA-204 може генерувати відкриті запитання чи підказки. Це може допомогти авторам розвинути свої навички генерувати питання та відповідати на них, а також організовувати та подавати свої ідеї в чіткій та стислій формі.

4. Мозковий штурм: генеруючи різноманітні запитання, пов’язані з конкретною темою, інструмент може стимулювати творче мислення та заохочувати досліджувати різні точки зору в окремих осіб або команд, які прагнуть створити нові ідеї чи вирішити проблеми.

5. Підготовка до співбесіди: QTA-204 можна використовувати, щоб допомогти людям відповідати на поширені запитання або створити власні запитання, які можна задати інтерв’юеру під час співбесіди при прийомі на роботи.

6. Дослідження ринку, зокрема маркетингові дослідження.. Компанії та організації можуть використовувати QTA-204 для створення запитань для досліджень ринку або фокус-груп. Це може допомогти їм краще зрозуміти вподобання та поведінку споживачів і приймати обґрунтовані рішення щодо продуктів, послуг або маркетингових стратегій.

7. Обслуговування клієнтів. Модель може допомогти представникам служби обслуговування клієнтів створювати персоналізовані запитання і відповіді на запити клієнтів. Це може покращити якість взаємодії з клієнтами та допомогти зміцнити довіру та лояльність клієнтів.

Модель також має потенціал повністю замінити людини в деяких сферах служби підтримки, наприклад при використанні у чат-ботах.

Крім того, залучення експертів, яке ми згадували у розділі 3.3, дає можливість для розробки нових та інтерактивних систем «людина в циклі». Зокрема, спочатку ми можемо використовувати QTA-204 для створення великої кількості запитань, потім ми можемо використати зворотний зв'язок щодо якості згенерованих запитань, наданих або експертами-людьми, або шляхом перевірки їхніх педагогічних цінностей для подальшого удосконалення можливостей QTA-204. Ці інтерактивні системи мають потенціал для вдосконалення зі збільшенням використання, що ідеально підходить для широкомасштабних освітніх програм.

Висновок до розділу 3

Таким чином, наша модель є результатом розробки автоматичного середовища для генерації тестів, що містять ряд питань. Модель буде розширено, щоб вона визначала певні методи для створення різних типів питань. Також варто зазначити, що для визначення типів запитань можна використовувати вже готові стандарти, наприклад QTI (IMS Question & Test Interoperability). Ця специфікація пропонує добре задокументований формат для розробки частин, необхідних для навчання та тестування систем електронного навчання. Використання такого стандарту для

визначення формату тестів і питань дає більше можливостей для роботи з різними елементами, такими як зберігання та обмін інформацією.

Ми також помітили, що попереднє навчання нерозміченим даним у домені може підвищити продуктивність у подальших завданнях (розділ 3.1). Цей висновок здебільшого ґрунтується на основних спостереженнях, таких як той факт, що SQuAD було створено з використанням даних з Вікіпедії. Було б корисно сформулювати більш точне поняття «подібності» між завданнями перед навчанням і подальшими завданнями, щоб ми могли зробити більш принциповий вибір щодо того, яке джерело немаркованих даних використовувати у майбутньому. Краще уявлення про пов’язаність завдань також зможе допомогти вибрати контрольовані завдання попереднього навчання.

Ми також зацікавлені в тому, щоб уникнути логістичних труднощів, пов’язаних із необхідністю завчасно вказувати, які мови може кодувати словник. Тому у подальших дослідженнях, QTA-204 можна модернізувати за допомогою мовно-агностичних методів, які дозволять їй формувати питання незалежно від мови тексту. Це особливо актуальне питання, оскільки англійська мова не є рідною для автора цього дослідження.

ВИСНОВКИ

Зростаючий інтерес до методів інтелектуального аналізу текстових даних призвів до поширення досліджень щодо їх функціонування та розвитку. Водночас глобальна економіка зазнає серйозних змін у зв'язку з появою нових форм економічних і соціальних відносин, що характеризуються виробництвом знань, інтеграцією технологій і розвитком децентралізованих інформаційних мереж. Ці мережі вимагають нових форм аналізу великої кількості інформації, що передається через них.

У цьому контексті нами було розроблено модель під назвою QTA-204, яка здатна автоматично генерувати запитання в різних форматах. Модель розроблено для створення безкоштовних, релевантних і «людських» запитань з більшим успіхом, ніж існуючі моделі. Крім того, запитання, створені QTA-204, є більш зрозумілими та релевантними контексту введення та відповідям, ніж ті, що створюються іншими моделями. QTA-204 здатний точно вловити тонкі відмінності між релевантною інформацією та створити відповідні запитання для кожної відповіді.

Щоб оцінити продуктивність цієї моделі, було використано Стенфордський набір даних відповідей на запитання (SQuAD), який включає понад 100 000 екземплярів даних, які складаються з абзацу, взятого зі статті у Вікіпедії, відповіді та запитання, створеного людиною на основі абзацу і відповісти. Ми розглядали абзац як вхідний контекст, а питання як вихідний, фактично перетворюючи SQuAD на навчальний набір даних для створення запитань. Індекси першого та останнього слів у відповіді надавались в наборі даних, що допомагало спростити кодування відповіді у відповідних векторах контекстних слів. Ми також скоротили кожен абзац до одного речення, що містить відповідь, і використовували це речення як контекст під час процесу навчання.

В результаті розроблений метод продемонстрував чудову продуктивність у порівнянні з кількому поширеними моделями (LSTM та Overgenerate & Rank) на

стандартному наборі даних SQuAD. Що ще важливіше, ми продемонстрували, що після навчання ми можемо адаптувати QTA-204 до навчального контенту та створювати плавні та релевантні питання.

Ці багатообіцяючі результати свідчать про те, що наша модель має потенціал до повної автоматизації та розширення процесу створення запитань для навчальних закладів, де потрібна велика кількість тестів, опитувань і практичних запитань, щоб супроводжувати рясний навчальний контент.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Єфіменко В. С. Автоматизоване тестування як метод педагогічної діагностики. *Theory and methods of e-learning*. 2014. Т. 4. С. 90–94. URL: https://doi.org/10.55056/e-learn.v4i1.375 (дата звернення: 25.12.2022).

2. Мироненко С., Онищенко Є. Порівняльний аналіз методів для вирішення задачі аналізу тексту. *Computer-integrated technologies: education, science, production*. 2020. № 40. С. 140–145. URL: https://doi.org/10.36910/6775-2524-0560-2020-40-21 (дата звернення: 25.12.2022).

3. Огляд методів обробки та аналізу текстів на природних мовах / С. І. Доценко та ін. *Інформаційно-керуючі системи на залізничному транспорті*. 2018. № 6. С. 26–32. URL: https://doi.org/10.18664/ikszt.v0i6.151638 (дата звернення: 25.12.2022).

4. Пентилюк М. Наукові параметри аналізу тексту. *Дивослово*. 2017. № 9 (726), верес. С. 36–41.

5. Серажим К. Інформаційний аспект аналізу тексту. *Журналістика*. 2008. Вип. 7 (32). С. 40–48.

6. Aithal S. G., Rao A. B., Singh S. Automatic question-answer pairs generation and question similarity mechanism in question answering system. *Applied intelligence*. 2021. URL: https://doi.org/10.1007/s10489-021-02348-9 (date of access: 25.12.2022).

7. Altenberg B. A bibliography of publications relating to English computer corpora. *English computer corpora*. Berlin, Boston. URL: https://doi.org/10.1515/9783110865967.355 (date of access: 25.12.2022).

8. A Study on Text Mining and Text Mining products. *International journal of science, technology and humanities*. 2014. Vol. 1, no. 1. P. 61–63. URL: https://doi.org/10.26524/ijsth11 (date of access: 25.12.2022).

9. Automatic question classifier / N. Patil et al. *2022 IEEE 4th international conference on cybernetics, cognition and machine learning applications (ICCCMLA)*, Goa, India, 8–9 October 2022. 2022. URL: https://doi.org/10.1109/icccmla56841.2022.9989066 (date of access: 25.12.2022).

10. Bacon D. IMS question and test interoperability. *MSOR connections*. 2003. Vol. 3, no. 3. P. 44–45. URL: https://doi.org/10.11120/msor.2003.03030044 (date of access: 25.12.2022).

11. Blake C. Text Mining. *Annual review of information science and technology*. 2011. Vol. 45, no. 1. P. 121–155. URL: https://doi.org/10.1002/aris.2011.1440450110 (date of access: 25.12.2022).

12. Exploring neural question generation for formal pragmatics: data set and model evaluation. *Frontiers*. URL: https://www.frontiersin.org/articles/10.3389/frai.2022.966013/full (date of access: 24.12.2022).

13. Finegan E., Johansson S. Computer corpora in english language research. *Language*. 1984. Vol. 60, no. 1. P. 190. URL: https://doi.org/10.2307/414219 (date of access: 25.12.2022).

14. Google trends. URL: https://trends.google.com/trends/ (date of access: 25.12.2022).

15. Intayoad W., Kamyod C., Temdee P. Reinforcement learning based on contextual bandits for personalized online learning recommendation systems. *Wireless personal communications*. 2020. Vol. 115, no. 4. P. 2917–2932. URL: https://doi.org/10.1007/s11277-020-07199-0 (date of access: 25.12.2022).

16. Interactivity within ims learning design and question and test interoperability. *3rd international conference on web information systems and technologies*, Barcelona, Spain, 3–6 March 2007. 2007. URL: https://doi.org/10.5220/0001269304400445 (date of access: 25.12.2022).

17. Joint learning of question answering and question generation / Y. Sun et al. *IEEE transactions on knowledge and data engineering*. 2020. Vol. 32, no. 5. P. 971–982. URL: https://doi.org/10.1109/tkde.2019.2897773 (date of access: 25.12.2022).

18. Karpicke J. D., Grimaldi P. J. Retrieval-Based learning: a perspective for enhancing meaningful learning. *Educational psychology review*. 2012. Vol. 24, no. 3. P. 401–418. URL: https://doi.org/10.1007/s10648-012-9202-2 (date of access: 25.12.2022).

19. Karpicke J. D., Roediger H. L. The critical importance of retrieval for learning. *Science*. 2008. Vol. 319, no. 5865. P. 966–968. URL: https://doi.org/10.1126/science.1152408 (date of access: 25.12.2022).

20. Kiran F., Gopal H., Dalvi A. Automatic question paper generator system. *International journal of computer applications*. 2017. Vol. 166, no. 10. P. 42–47. URL: https://doi.org/10.5120/ijca2017914138 (date of access: 25.12.2022).

21. Last M., Danon G. Automatic question generation. *WIREs data mining and knowledge discovery*. 2020. Vol. 10, no. 6. URL: https://doi.org/10.1002/widm.1382 (date of access: 25.12.2022).

22. Memory-Efficient learning for large-scale computational imaging / M. Kellman et al. *IEEE transactions on computational imaging*. 2020. Vol. 6. P. 1403–1414. URL: https://doi.org/10.1109/tci.2020.3025735 (date of access: 25.12.2022).

23. Neural question generation from text: a preliminary study / Q. Zhou et al. *Natural language processing and chinese computing*. Cham, 2018. P. 662–671. URL: https://doi.org/10.1007/978-3-319-73618-1_56 (date of access: 25.12.2022).

24. NLP-progress repository. *NLP-progress*. URL: https://nlpprogress.com/english/question_answering.html (date of access: 24.12.2022).

25. OpenStax | free textbooks online with no catch. OpenStax. URL: https://openstax.org/ (date of access: 25.12.2022).

26. Ostrovska K. Дослідження методів інтелектуального аналізу даних для обробки результатів тестування. *System technologies*. 2020. Т. 4, № 129. С. 146–159. URL: https://doi.org/10.34185/1562-9945-4-129-2020-14 (дата звернення: 25.12.2022).

27. Rohrer D., Pashler H. Recent research on human learning challenges conventional instructional strategies. *Educational researcher*. 2010. Vol. 39, no. 5. P. 406–412. URL: https://doi.org/10.3102/0013189x10374770 (date of access: 25.12.2022).

28. Solving the SQuAD Problem. *brett koonce*. URL: https://brettkoonce.com/talks/solving-the-squad-problem/ (date of access: 24.12.2022).

29. SQuAD: 100,000+ questions for machine comprehension of text. *arXiv.org e-Print archive*. URL: https://arxiv.org/pdf/1606.05250.pdf (date of access: 24.12.2022).

30. The stanford question answering dataset. *GitHub Pages*. URL: https://rajpurkar.github.io/SQuAD-explorer/ (date of access: 24.12.2022).

31. Tom M. D., Tenorio M. F. A neural computation model with short-term memory. *IEEE transactions on neural networks*. 1995. Vol. 6, no. 2. P. 387–397. URL: https://doi.org/10.1109/72.363474 (date of access: 25.12.2022).

32. Wei J. The quick guide to squad. *Medium*. URL: https://towardsdatascience.com/the-quick-guide-to-squad-cae08047ebee (date of access: 24.12.2022).

33. Wichmann A. A study of up-arrows in the lancaster/ibm spoken english corpus. *English computer corpora*. Berlin, Boston. URL: https://doi.org/10.1515/9783110865967.165 (date of access: 25.12.2022).

34. Wolfe J. H. Automatic question generation from text - an aid to independent study. *ACM SIGCSE Bulletin*. 1976. Vol. 8, no. 1. P. 104–112. URL: https://doi.org/10.1145/952989.803459 (date of access: 25.12.2022).

35. Woo S., Li Z., Mirkovic J. Good automatic authentication question generation. *Proceedings of the 9th international natural language generation conference*, Edinburgh, UK. Stroudsburg, PA, USA, 2016. URL: https://doi.org/10.18653/v1/w16-6632 (date of access: 25.12.2022).

36. Xie Q. Corpus linguistics and corpus-based research in hong kong: a state-of-art review. *English language and literature studies*. 2013. Vol. 3, no. 3. URL: https://doi.org/10.5539/ells.v3n3p48 (date of access: 25.12.2022).

Додаток А

Доповідь до дипломної роботи

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра інженерії програмного забезпечення

МАГІСТЕРСЬКА РОБОТА
«РОЗРОБКА МЕТОДИКИ АВТОМАТИЗОВАНОЇ ПОБУДОВИ ОПИТУВАЛЬНИКА НА ОСНОВІ КОНСПЕКТУ ЛЕКЦІЙ З ВИКОРИСТАННЯМ МЕТОДІВ TEXT MINING»

Виконав: студент групи ПДМ – 61, Кононенко Ілля Віталійович

Керівник: к.ф-м.н., доцент кафедри Інженерії програмного забезпечення Садовенко Володимир Сергійович

Київ - 2023

АКТУАЛЬНІСТЬ ДОСЛІДЖЕННЯ

Актуальність даного дослідження полягає у наступному:

- Зростання інтересу до методів інтелектуального аналізу тексту;
- Впровадження нових форм аналізу чисельних потоків інформації у сферах бізнесу;
- Необхідність трансформації наявних підходів до аналізу текстових даних через надмірну складність процесу аналізу, структуру природної мови та джерел інформації;
- В останні роки, електронні тести стали переважаючим методом іспити в українських університетах.

3

МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

Мета роботи: розробити нову методику автоматизованої побудови опитувальника та модернізувати наявні методи використання алгоритмів Text Mining для майбутніх досліджень.

Об'єкт дослідження: використання методів Text Mining для автоматизованої побудови опитувальника.

Предмет дослідження: обмеження алгоритму побудови автоматизованого опитувальника на основі методів Text Mining.

4

ЗАВДАННЯ ДОСЛІДЖЕННЯ

Завданням дослідження є:

1. Проаналізувати предметну область інтелектуального аналізу тексту, її завдання та застосування у системах автоматичної побудови з існуючого тексту.
2. Дослідити методи інтелектуального аналізу тексту, його архітектуру та компоненти.
3. Визначити методи та засоби побудови системи вилучення даних, розробити метод автоматизованої побудови з використанням SQuAD як корпусу.
4. Проаналізувати ефективність розробленого методу у порівнянні з базовими рівнями Overgenerate & Rank та LTSM.

5

TEXT MINING

Text Mining (Інтелектуальний аналіз тексту) – процес вилучення цікавих і нетривіальних шаблонів або знань із текстових документів. Його походження – поєднання різних суміжних галузей:

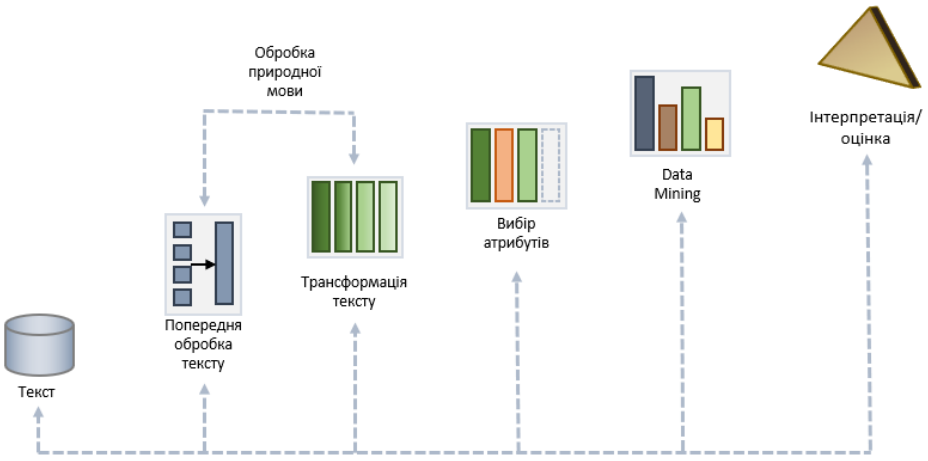
- Data Mining (Інтелектуальний аналіз даних);
- Штучний інтелект;
- Статистика;
- Управління базами даних;
- Бібліотекознавство;
- Лінгвістика.

Text Mining можна використовувати для різних областей: від базових описів текстового вмісту через підрахунок слів до більш складних способів, такими як пошук зв'язків між авторами та оцінка вмісту сценарії.

6

TEXT MINING: МЕТА ТА ПРОЦЕС

Мета Text Mining – обробка неструктурованої інформації, що міститься в текстових даних, щоб зробити текст доступним для різних статистичних алгоритмів Data Mining.



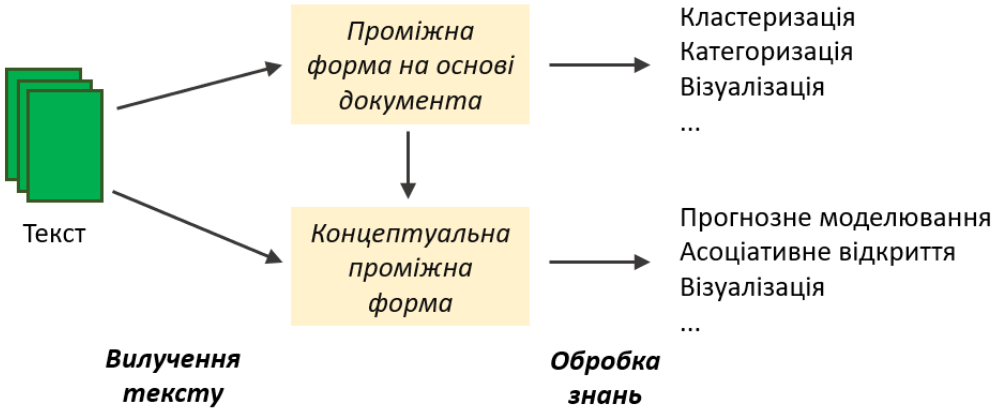
Процес Text Mining

ЗАГАЛЬНА СТРУКТУРА TEXT MINING

7

- **Уточнення (вилучення) тексту** – перетворює текстові документи вільної форми на проміжну;

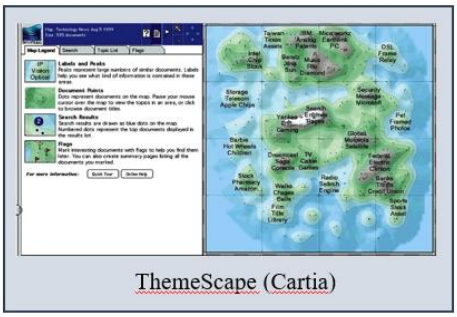
- **Дистиляція (обробка) знань** – виводить моделі або знання з проміжної форми.



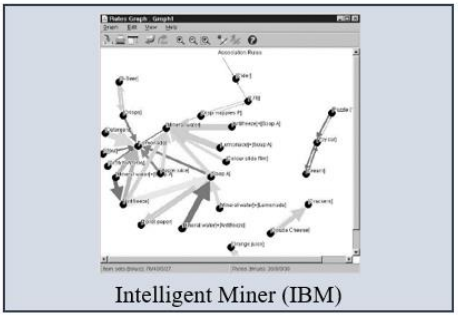
Фреймворк Text Mining

ПРИКЛАДИ ІСНУЮЧИХ ІТ-РІШЕНЬ ТА ЇХ МОДЕЛЕЙ

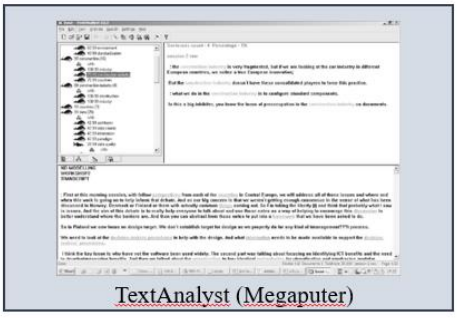
8



ThemeScape (Cartia)



Intelligent Miner (IBM)



TextAnalyst (Megaputer)

9

SQUAD ЯК КОРПУС АВТОМАТИЗОВАНОЇ ПОБУДОВИ

SQuAD (Stanford Question Answering Dataset) – набір даних про розуміння прочитаного, що складається із запитань, поставлених краудворкерами щодо набору статей Вікіпедії, де відповіддю на кожне запитання є фрагмент тексту або проміжок із відповідного уривку для читання, інакше питання може бути неможливим.

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under **gravity**. The main forms of precipitation include drizzle, rain, sleet, snow, **graupel** and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals **within a cloud**. Short, intense periods of rain in scattered locations are called "showers".

What causes precipitation to fall?
gravity

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail?
graupel

Where do water droplets collide with ice crystals to form precipitation?
within a cloud

Зразок пар запитань і відповідей для уривка з набору даних SQuAD

10

ОСОБЛИВОСТІ SQUAD

У SQuAD можна виділити кілька особливостей:

- SQuAD є великим, бо містить великі набори даних (понад 100 тис. запитань);
- SQuAD є складним: пропущення відповіді вже не стає таким пробачливим;
- SQuAD вимагає аргументації.

SQuAD є одним із найпопулярніших наборів даних із відповідями на запитання, який використовують у дослідженнях з метою оцінки моделі обробки природної мови.

11

МОДЕЛЬ QTA-204 ЯК РОЗРОБЛЕНА МОДЕЛЬ

QTA-204 – керована даними модель генерації запитань, яка слідує загальній структурі читача-генератора, але використовує декілька ключових нововведень для збереження плавності та доцільності згенерованих питань.

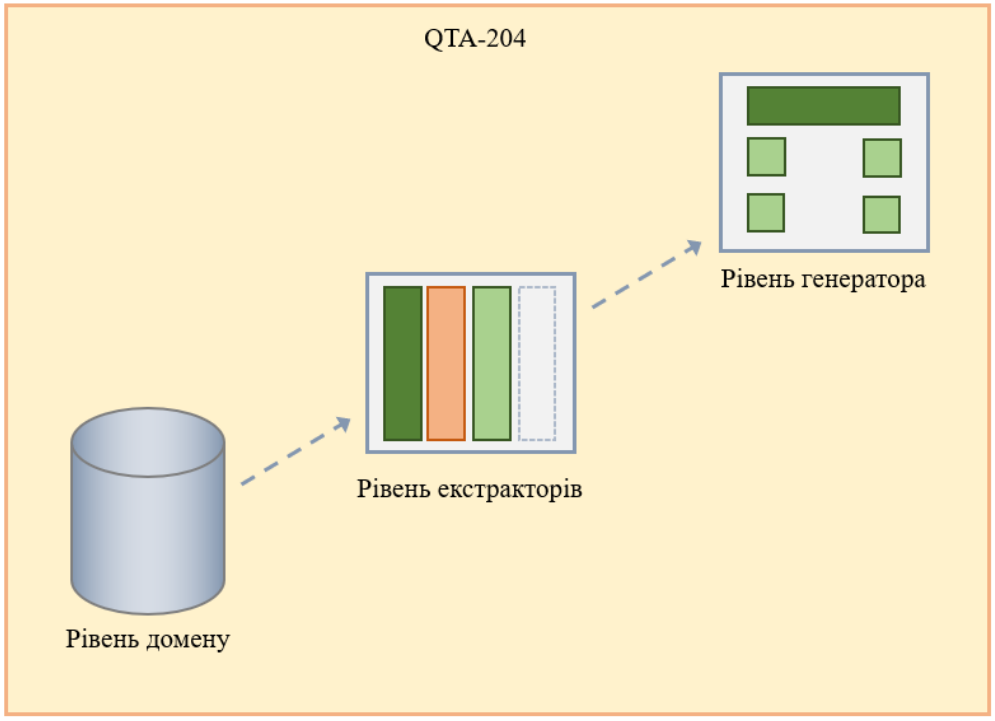
Підхід є здебільшого універсальним і підходить до створення різних типів питань структурованого навчального процесу.

Підхід моделі складається з трьох кроків:

- Структуроване представлення знань, включених до навчальної програми;
- Вилучення окремих компонентів, створення структури вмісту;
- Створення запитань з даних, отриманих під час попереднього кроку.

12

ФОРМАЛЬНА МОДЕЛЬ ФОРМУВАННЯ ЗАПИТАНЬ



Формальна модель формування запитань

13

ФОРМАЛЬНА МОДЕЛЬ ФОРМУВАННЯ ЗАПИТАНЬ: РІВЕНЬ ДОМЕНУ

Основні комплекти моделі:

- A, AS, AE, AR, AD – набір аксіом, що описують область де:
  – AS: набір аксіом типу Sub Class Of;
  – AE : набір аксіом типу еквівалентних класів;
  – AR : набір аксіом типу Діапазон властивостей об'єкта;
  – AD : набір аксіом типу Непересічні класи;
- C: набір понять.
- P: набір властивостей.
- R: набір обмежень.
- Ann: набір анотацій.

Визначення наборів і структур, пов'язаних із доменом (рівень домену)

14

ФОРМАЛЬНА МОДЕЛЬ ФОРМУВАННЯ ЗАПИТАНЬ: РІВЕНЬ ЕКСТРАКТОРІВ

$$\overrightarrow{h_j} = \text{bi} - \text{LSTM}(\tilde{c}_j, \overrightarrow{h_{j-1}})$$

Формула прихованого стану, що відповідає прямому напрямку

Набори:

- Axi – набір обраних аксіом за певним критерієм;
- K – набір критеріїв вибору аксіом;
- T {tI,tD} – набір типів речень, який складається з двох елементів, де:
  - tI – запитальний тип;
  - tD – декларативний тип.

Екстрактори:

- selectAxi: A → Axi – вибирає аксіоми, щоб ідентифікувати набір аксіом за певний критерій;
- extrBaseConcept: A → C – виділяє базове поняття з аксіоми;
- extrProperty: A → 2
- P – виділяє властивість з аксіоми;
- extrRestrConcept: A → C – витягує набір обмежувальних понять із обмеження в аксіомі;
- extrRange: P → C – витягує концепцію Діапазон властивості;
- extrAnnotations: A, P, C
  - Ann – витягує анотації сутності або аксіоми зі структури знань;
  - extrDisjointConcepts: C → C – виділяє непересічні класи.

Визначення наборів та структур, пов'язаних з екстракторами

ФОРМАЛЬНА МОДЕЛЬ ФОРМУВАННЯ ЗАПИТАНЬ: РІВЕНЬ ГЕНЕРАТОРА

15

$$P(Q|C, A, \theta) = \prod_{t=1}^{L} P(q_t|C, A, q_\tau{}_{\tau=1}^{t-1}, \theta)$$

Формула генерації питання (ітеративна вибірка питальних слів)

Набори:

- Q – набір питань.

Крім того, існують два типи питань:

- Q – набір класичних запитань;
- QD – набір декларативних питань.

- Ans: набір відповідей.

Визначення наборів генераторів

АНАЛІЗ ЕФЕКТИВНОСТІ РОЗРОБЛЕНОГО МЕТОДУ ЗА ПЕРЕКЛАДОМ

16

Модель	Метрика		
	BLEU	METEOR	ROUGE-L
Overgenerate & Rank	0.1120	0.1702	0.2792
LSTM	0.0231	0.0796	0.2703
QTA-204	0.1386	0.1838	0.4437

Порівняння між QTA-204, LSTM та Overgenerate & Rank

Оцінка відбувалася за трьома метриками:

- BLEU – подібність тексту машинного перекладу до набору високоякісних довідкових перекладів;
- METEOR – оцінка результатів машинного перекладу;
- ROUGE-L – оцінка автоматичного підведення підсумків текстів, а також машинних перекладів на основі найдовшої спільної підпослідовності.

АНАЛІЗ ЕФЕКТИВНОСТІ РОЗРОБЛЕНОГО МЕТОДУ: ПРИКЛАДИ ПИТАНЬ, ЗГЕНЕРОВАНИХ МОДЕЛЯМИ

17

Модель	Метрика		
	10%	50%	100%
Overgenerate & Rank	When Albert Einstein was nominated for a Nobel Prize in Physics?	When Albert Einstein was nominated for a Nobel Prize in Physics?	When Albert Einstein was nominated for a Nobel Prize in Physics?
LSTM	What is nominated for a Nobel Prize?	In what year Albert Einstein was nominated for a Nobel Prize?	In what year Albert Einstein was nominated for a Nobel Prize?
QTA-204	When was Einstein nominated for a Nobel Prize?	When was Einstein nominated for a Nobel Prize?	When was Einstein nominated for a Nobel Prize?

Приклади питань, згенерованих QTA-204, LSTM та Overgenerate & Rank

Продуктивність підвищується з більшою кількістю навчальних даних. Навчання моделі на 10%, 50% і 100% навчального набору SQuAD.

18

ВИСНОВКИ

Мета роботи розробки нової методики автоматизованої побудови опитувальника та модернізування наявних методів використання алгоритмів Text Mining для майбутніх досліджень досягнута. В роботі:

1. Проаналізовано предметну область інтелектуального аналізу тексту, її завдання та застосування у системах автоматичної побудови з існуючого тексту.
2. Досліджено методи інтелектуального аналізу тексту, його архітектуру та компоненти.
3. Визначено методи та засоби побудови системи вилучення даних, розроблено метод автоматизованої побудови з використанням SQuAD як корпусу.
4. Проаналізовано ефективність розробленого методу у порівнянні з базовими рівнями Overgenerate & Rank та LTSM.

ПУБЛІКАЦІЇ ТА АПРОБАЦІЯ РОБОТИ

Тези доповідей:

1. Кононенко І.В., Методологічні засади використання природної мови у процесі проведення інтелектуального аналізу тексту. // Науково-практична інтернет-конференція «Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення (випуск 73)». – Тернопіль: Оргкомітет МНІК «Конференція онлайн», 2022.
2. Кононенко І.В., Садовенко В.С., Про застосування алгоритмів text mining для аналізу даних про настроїв аудиторії соціальних мереж (на прикладі Twitter) // Міжнародна мультидисциплінарна інтернет-конференція «Світ наукових досліджень. Випуск 15». – Тернопіль: Оргкомітет ММНІК , 2022.

ДЯКУЮ ЗА УВАГУ!