Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра системного аналізу та інформаційних технологій

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

“Інформаційна інтелектуальна технологія автоматичного синтезу анотацій до матеріалів про водні ресурси”

Виконав: студент 2 курсу, групи 2ІСТ-21м
спеціальності 126 – «Інформаційні системи та технології»
_____ Хмарук С.М.

Керівник: д.т.н., проф. каф. САІТ
_____ Мокін В.Б.
«01» 12 2022 р.

Опонент: к.т.н., доц. каф. АІІТ
_____ Сторчак В.Г.
«15» 12 2022 р.

Допущено до захисту

Завідувач кафедри САІТ
_____ д.т.н., проф. Мокін В. Б.
«05» 12 2022 р.

Вінниця ВНТУ – 2022 рік

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра системного аналізу та інформаційних технологій
Рівень вищої освіти – ІІ-й (магістерський)
Галузь знань – 12 Інформаційні технології
Спеціальність – 126 Інформаційні системи та технології
Освітньо-професійна програма – Інформаційні технології аналізу даних та зображень

ЗАТВЕРДЖУЮ
Завідувач кафедри САІТ
_____ д.т.н., проф. Мокін В. Б.
«16» 09 2022 р.

ЗАВДАННЯ
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ
Хмаруку Сергію Михайловичу

1. Тема роботи: "Інформаційна інтелектуальна технологія автоматичного синтезу анотацій до матеріалів про водні ресурси",
керівник роботи: Мокін В.Б., д.т.н., проф. каф. САІТ,
затверджені наказом закладу вищої освіти від «14» 09 2022 року №203
2. Строк подання студентом роботи «01» 12 2022 року
3. Вихідні дані до роботи:
Датасет Kaggle «NLP : Reports & News Classification» з даними про водні ресурси.
4. Зміст текстової частини:
   - характеристика проблеми автоматичного синтезу анотацій до матеріалів про водні ресурси;
   - основні етапи виконання роботи та автоматичні екологічні звіти і класифікація новин;
   - створення інтелектуальної інформаційної технології автоматичного синтезу анотацій до матеріалів про водні ресурси методами штучного інтелекту;
   - економічна частина.
5. Перелік ілюстративного матеріалу:
   - алгоритм роботи програми;
   - загальновживані слова;
   - хмари слів;
   - середня довжина слів;
   - результати анотування текстів.

6. Консультанти розділів МКР

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
4	Бурєннікова Н.В, д.е.н., проф. каф. ЕПВМ	10.11.22 р.	25.11.22 р.

7. Дата видачі завдання «16» 09 2022 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів МКР	Строк виконання етапів роботи	Примітка
1	Аналіз предметної області	09.2022	
2	Вибір оптимальних технологій та проведення розвідувального аналізу	10.2022	
3	Розроблення інтелектуальної інформаційної технології	11.2022	
4	Економічна частина	11.2022	
5	Оформлення матеріалів до захисту МКР	11.2022	

Студент _____ Хмарук С. М.

Керівник роботи _____ Мокін В. Б.

АНОТАЦІЯ

УДК 004.08+004.93

Хмарук С.М. Інформаційна інтелектуальна технологія автоматичного синтезу анотацій до матеріалів про водні ресурси. Магістерська кваліфікаційна робота зі спеціальності 126 – інформаційні системи та технології, освітньо-професійна програма – інформаційні технології аналізу даних та зображень. Вінниця: ВНТУ, 2022. 108 с.

На укр. мові. Бібліогр.: 42 назв; рис.: 67; табл.: 7.

В магістерській кваліфікаційній роботі розроблено концепцію інформаційної інтелектуальної технології, яка забезпечить автоматичний синтез анотацій до матеріалів про водні ресурси. Вибрано оптимальні технології та за їх допомогою створено пілотну версію програми, роботу програми випробувано на авторському україно-англомовному датасеті для басейну р. Південний Буг.

Ілюстративна частина складається з 8 плакатів із результатами розвідувального аналізу та анотування тексту методами NLP.

У розділі економічної частини розглянуто питання про доцільність розробки та впровадження інформаційної технології анотування тексту методами штучного інтелекту.

Ключові слова: NLP, інформаційна технологія, інтелектуальна технологія, проблеми навколишнього середовища, геокодування, класифікація тексту, синтез тексту.

ABSTRACT

Khmaruk S.M. Information intelligent technology of automatic synthesis of annotations to materials about water resources. Master's qualification thesis on specialty 126 - information systems and technologies, educational and professional program - information technologies of data and image analysis. Vinnytsia: VNTU, 2022. 108 p.

In Ukrainian speech Bibliography: 42 titles; Fig.: 67; tab.: 7.

In the master's qualification work, the concept of information intelligent technology was developed, which will provide automatic synthesis of annotations to materials about water resources. Optimal technologies were selected and a pilot version of the program was created with their help, the program was tested on the author's Ukrainian-English dataset for the basin of the Southern Bug River.

The illustrative part consists of 8 posters with the results of intelligence analysis and text annotation using NLP methods.

In the section of the economic part, the issue of the feasibility of developing and implementing the information technology of text annotation using artificial intelligence methods is considered.

Keywords: NLP, information technology, intelligent technology, environmental problems, geocoding, text classification, text synthesis.

ЗМІСТ

ВСТУП

Актуальність теми. На сьогоднішній день в багатьох областях науки і суспільного життя на машини покладаються ті завдання, які раніше була в змозі виконувати лише людина і саме це обумовило створення штучного інтелекту і надалі стимулює його розвиток та удосконалення. Метою створення штучного інтелекту, першочергово, є поліпшення життя людини. Проте, будь-яка система має свої недоліки і проблеми, які необхідно досліджувати для подальшого її вдосконалення та ефективного розвитку.

Резюмування тексту — це процес створення коротшого тексту без усунення семантичної структури тексту. Резюмування тексту є дуже корисною та важливою частиною обробки природної мови (NLP).

З усіх завдань обробки природної мови (NLP) конспектування, мабуть, є одною із найменших проблем, тому що скорочення вмісту статті є набагато менш приголомшливим, ніж те, що GPT-3 автоматично генерує ідеї для стартапів. Однак, незважаючи на низький профіль, резюмування тексту далеке від вирішення, особливо в промисловості.

Метою дослідження є підвищення адекватності анотацій текстовим матеріалам про водні ресурси, які синтезуються автоматично з використанням методів штучного інтелекту.

Розробка системи передбачає виконання наступних задач:

– вибір оптимальних технологій;

– розвідувальний аналіз даних;

– побудова NLP-моделей.

Об’єктом дослідження магістерської кваліфікаційної роботи є автоматичний синтез анотацій до матеріалів про водні ресурси.

Предметом дослідження магістерської кваліфікаційної роботи є інтелектуальна інформаційна технологія автоматичного синтезу анотацій до матеріалів про водні ресурси.

Методи дослідження. У дослідженнях було використано методи і технології розвідувального аналізу даних та машинного навчання, у т.ч. методи NLP-технологій.

Новизна одержаних результатів. Наукова новизна полягає в тому, що дістала подальший розвиток інтелектуальна інформаційна технологія автоматичного синтезу анотацій до матеріалів про водні ресурси за рахунок підвищення адекватності цих анотацій.

Апробація результатів магістерської кваліфікаційної роботи. Результати роботи доповідались на Всеукраїнській науково-практичній інтернет-конференції «Молодь в науці: дослідження, проблеми, перспективи» (Вінниця, 2021-2022 рр.).

Публікації результатів магістерської кваліфікаційної роботи. Опубліковано тези на Всеукраїнській науково-практичній інтернет-конференції «Молодь в науці: дослідження, проблеми, перспективи» (Вінниця, 2021-2022 рр.) [1].

1 ХАРАКТЕРИСТИКА ПРОБЛЕМИ АВТОМАТИЧНОГО СИНТЕЗУ АНОТАЦІЙ ДО МАТЕРІАЛІВ ПРО ВОДНІ РЕСУРСИ

1.1 Суть проблеми

Штучний інтелект (ШІ) (artificial intelligence – AI) – властивість автоматичних систем брати на себе окремі функції інтелекту людини, наприклад, вибирати й ухвалювати оптимальні рішення на основі раніше одержаного досвіду й раціонального аналізу зовнішніх дій [2].

Резюмування - це підведення підсумків. Суть цього прийому слухання в тому, що ми своїми словами підводимо підсумок основним думкам співрозмовника. Резюмуюча фраза - це мова співрозмовника в «згорненому» вигляді, її головна ідея.

Резюмування (анотування) є одним із найпоширеніших актів мовної поведінки. Якщо нас просять розповісти, що сталося на зустрічі, що хтось розповів нам про іншу особу чи подію, про що була телевізійна програма, нас запитують або запрошують виражати в стислій формі основні частини попереднього усного чи письмового тексту чи дискурсу. Є багато людей, чия повсякденна робота полягає значною мірою, якщо не виключно, у створенні узагальнень або резюме: журналіст, який повідомляє про результати розслідування; суддя, який підсумовує подані в суді докази; державний службовець, який проводить огляд аргументів за та проти конкретної пропозиції; секретар товариства, який веде протокол засідання правління; вчений, який переглядає останні публікації у своїй галузі досліджень і так далі.

Резюмування скорочує довший текст до коротшого, відкидаючи при цьому менш важливу інформацію (рис. 1.1).

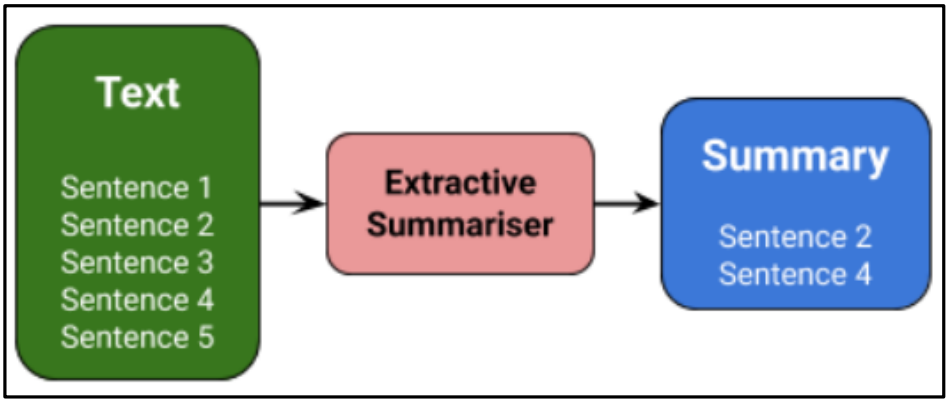

Рисунок 1.1 – Схема анотування

Резюмування тексту є дуже корисною та важливою частиною обробки природної мови (NLP). Спочатку поговоримо про те, що таке реферування тексту. Припустімо, що у нас забагато рядків текстових даних у будь-якій формі, наприклад зі статей, журналів чи соціальних мереж. У нас брак часу, тому ми хочемо лише короткий звіт цього тексту. Ми можемо підсумувати наш текст у кілька рядків, видаливши неважливий текст і перетворивши той самий текст у меншу семантичну текстову форму.

Тепер давайте подивимося, як можна застосувати NLP у програмуванні. Пізніше розглянемо всі підходи, а тут класифікуємо підходи NLP.

У цьому підході створюються алгоритми або програми, які зменшують розмір тексту та створюють зведення наших текстових даних. У машинному навчанні це називається автоматичним підсумовуванням тексту [3].

Резюмування тексту — це процес створення коротшого тексту без усунення семантичної структури тексту (рис. 1.2).

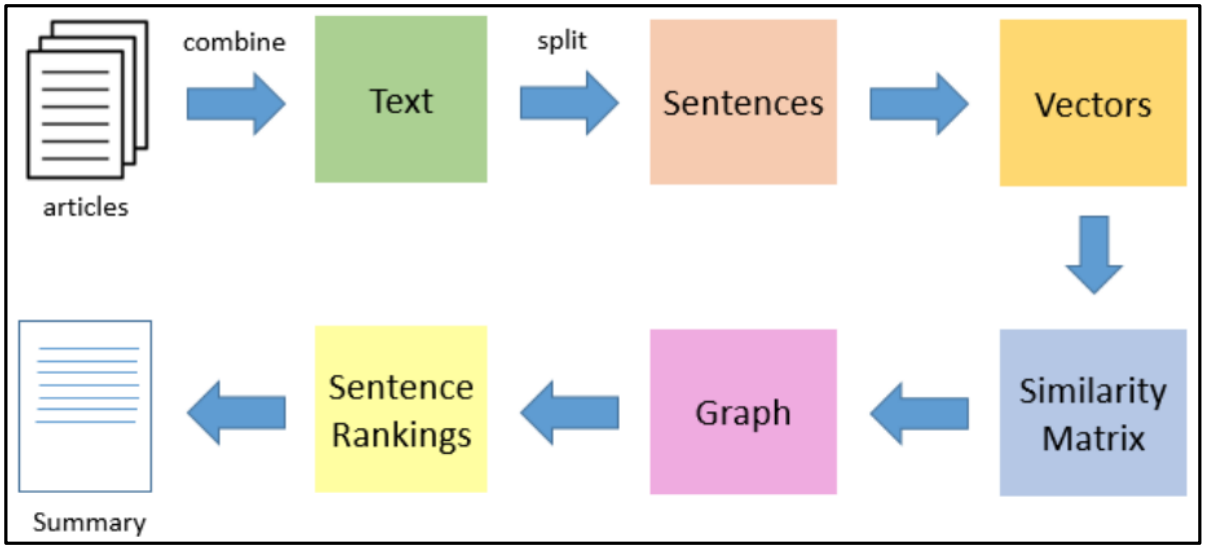

Рисунок 1.2 – Етапи створення анотування [3]

З усіх завдань обробки природної мови (NLP) конспектування, мабуть, є одним із найменших заголовків. Скорочення вмісту статті є набагато менш приголомшливим, ніж те, що GPT-3 автоматично генерує ідеї для стартапів. Однак, незважаючи на низький профіль, резюмування тексту далеке від вирішення, особливо в промисловості. Рудиментарні API, надані великими іменами, такими як Microsoft, залишають достатньо місця для невеликих компаній для узагальнення з різних точок зору, поки що явного переможця не видно.

Резюмування тексту можна виконати двома способами:

– екстракційний метод це – вибір n номерів найважливіших речень зі статті, які, швидше за все, передають зміст статті. Цей підхід дуже простий у застосуванні та зручний для початківців. Це – головна причина вибору цього методу;

– абстрактний метод це – метод, який використовує такі концепції глибокого навчання, як архітектура кодера-декодера, мережі LSTM (довго короткострокової пам’яті), які дуже важко зрозуміти новачкам. Цей метод генерує повністю нове резюме статті та містить речення, яких навіть немає в оригінальній статті. Тобто він може призвести до створення речень, які взагалі не мають жодного значення.

Одна з проблем полягає в тому, що людство досі не може точно окреслити межі того, що воно позначає терміном «штучний інтелект» і, зрештою, як і те, кого чи що можна або слід називати «роботами». Так, поступово, межа між здібностями людей і носіями так званого штучного інтелекту розмивається [4].

В наш час люди публікують безліч текстової інформації, статті, публікації та інші документи. І потім власноруч пишуть тези (анотації) до них. Тому було прийнято рішення це робити методами штучного інтелекту, а саме – використовуючи певні моделі розробити систему, яка буде автоматично обирати головне з тексту екстракційним методом.

Для прикладу візьмемо текст з географічними назвами (рис. 1.3) та проведемо синтез тексту (рис. 1.4) за допомогою коду одного з грандмайстрів Kaggle [5].

```
Text blocks:
 ['The Southern Bug River is the only large river in Ukraine, whose catchment area
is fully located within the country, which allows developing River Basin Management
Plan and Programme of Measures to implement it as national, without any external in
fluence and interstate limitations. Overregulation, developed agrarian sector, a nu
mber of large cities, energy and industrial installations cause significant man-mad
e load at the river. One of its main consequences is enrichment of the water by nut
rients, especially nitrogen and phosphorus compounds and significant eutrophicatio
n. In the same time, there are undisturbed sites in the basin, where unique flora a
nd fauna of Podillya, granite-steppe Pobuzhya and lowlands near the Black Sea is pr
eserved. In the middle and end of the last century, development of energy sector ca
used significant interest to the Southern Bug in view of its hydrochemical and hydr
obiological state as a source of water supply for Ladyzhin heat electrostation. In
the first years of Ukrainian independence, in frame of the State program “Restorati
on of surface water quality in Ukraine”, the Southern Bug basin was selected as a p
ilot for the development of modern environmental water monitoring. ']
```

Рисунок 1.3 – Текст з географічними назвами

```
BERT Summarizing Result:
The Southern Bug River is the only large river in Ukraine, whose catchment area is
fully located within the country, which allows developing River Basin Management Pl
an and Programme of Measures to implement it as national, without any external infl
uence and interstate limitations. In the first years of Ukrainian independence, in
frame of the State program “Restoration of surface water quality in Ukraine”, the S
outhern Bug basin was selected as a pilot for the development of modern environment
al water monitoring.

GPT-2 Summarizing Result:
The Southern Bug River is the only large river in Ukraine, whose catchment area is
fully located within the country, which allows developing River Basin Management Pl
an and Programme of Measures to implement it as national, without any external infl
uence and interstate limitations. In the middle and end of the last century, develo
pment of energy sector caused significant interest to the Southern Bug in view of i
ts hydrochemical and hydrobiological state as a source of water supply for Ladyzhin
heat electrostation.

XLNet Summarizing Result:
The Southern Bug River is the only large river in Ukraine, whose catchment area is
fully located within the country, which allows developing River Basin Management Pl
an and Programme of Measures to implement it as national, without any external infl
uence and interstate limitations. In the middle and end of the last century, develo
pment of energy sector caused significant interest to the Southern Bug in view of i
ts hydrochemical and hydrobiological state as a source of water supply for Ladyzhin
heat electrostation.
```

Рисунок 1.4 – Результат синтезу тексту

Моделі GPT-2 та XLNet показали однаковий результат, визначивши основним частину з власною назвою населеного пункту, при тому що BERT-модель визначила основним текст з цитатою. В даному тексті більш головним буде саме частина з цитатою, тому що це є назвою державної програми, яку було обрано для розвитку.

1.2 Огляд існуючих методів вирішення проблеми

Обробка природної мови (Natural Language Processing, NLP) — загальний напрям інформатики, штучного інтелекту й математичної лінгвістики. Він вивчає проблеми комп’ютерного аналізу й синтезу природної мови. Щодо штучного інтелекту, то аналіз — це розуміння мови, а синтез — генерація розумного тексту.

Методи NLP дозволяють вирішувати різноманітні завдання: від класифікації тональності текстів, машинного перекладу, спам-фільтрів, розмітки частин мови до генерації тексту [6].

Обробка природної мови може допомогти вам у виконанні багатьох завдань, і сфери застосування просто збільшуються щодня (рис. 1.5).

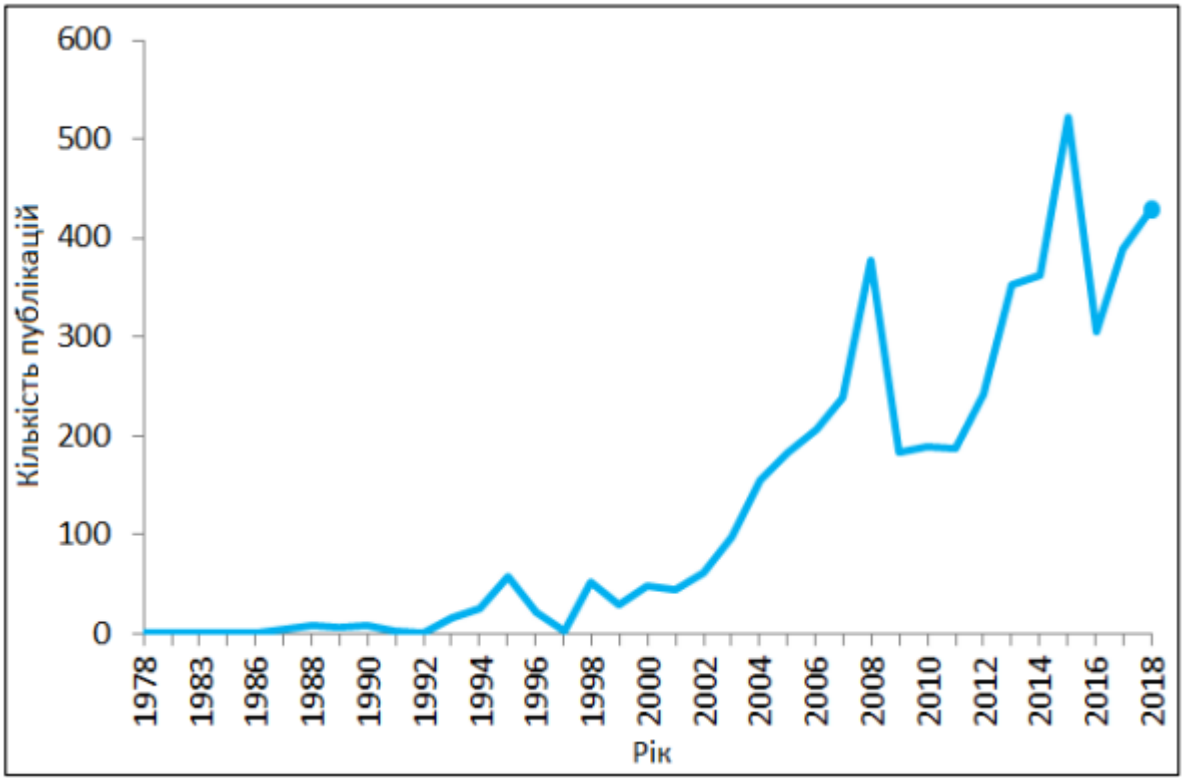

Рисунок 1.5 – Графік зростання кількості згадок про NLP в публікаціях

Summarization – вираження найважливіших фактів чи ідей про щось або когось у короткій і зрозумілій формі або текст, у якому ці факти чи ідеї виражені.

Summarization є одним із найпоширеніших завдань обробки природної мови (NLP). З огляду на кількість нового контенту, який щодня створюють мільярди людей та їхні смартфони, ми щодня завалені все більшою кількістю даних. Люди можуть споживати лише обмежену кількість інформації, і їм потрібен спосіб відфільтрувати і знайти інформацію, яка має значення. Резюмування тексту може допомогти досягти цього для текстової інформації.

Можемо відокремити сигнал від шуму та вжити значущих дій щодо них [7] (рис. 1.6).

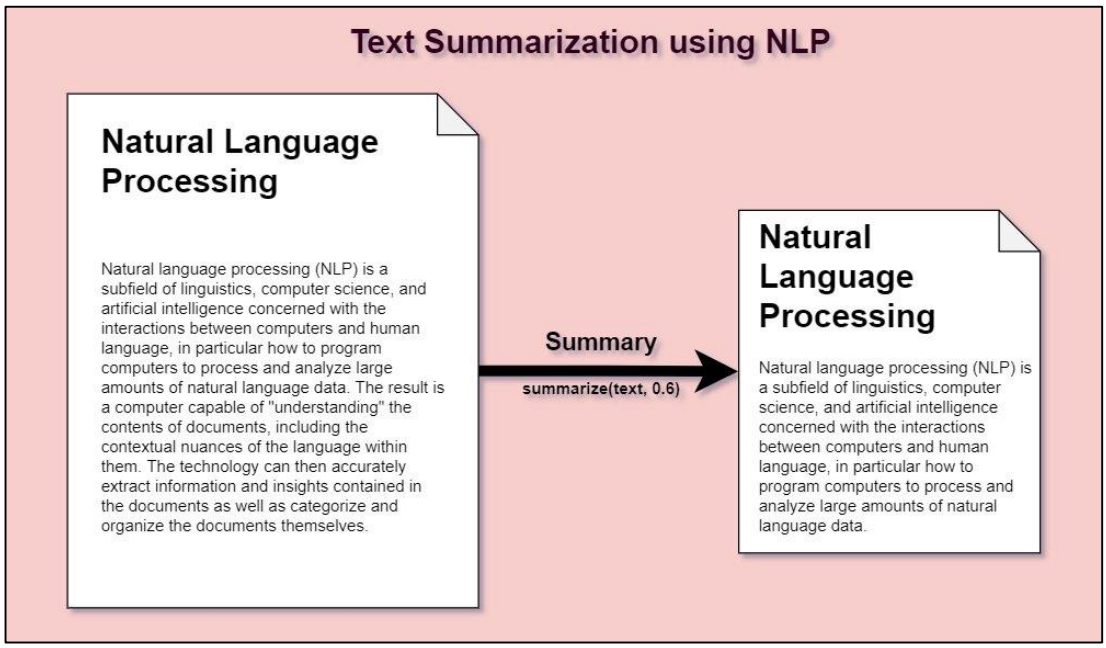

Рисунок 1.6 – Анотування тексту [7]

Може виникнути спокуса використовувати реферування для всіх текстів, щоб отримати з них корисну інформацію та витрачати менше часу на читання. Однак на даний момент узагальнення NLP було успішним випадком використання лише в кількох сферах.

Резюмування тексту чудово працює, якщо текст містить багато необроблених фактів і може бути використаний для фільтрації важливої інформації з них. Моделі NLP можуть узагальнювати довгі документи та представляти їх у невеликих простіших реченнях. Новини, інформаційні бюлетені та розсилки підпадають під ці категорії.

Однак для текстів, де кожне речення спирається на попереднє, підсумовування тексту працює не дуже добре. Дослідницькі журнали, медичні тексти є гарними прикладами текстів, де резюмування може бути не дуже вдалим.

Отже, підсумовування тексту корисне лише в кількох випадках використання [7].

Існує два основних типи резюмування тексту (суммарайзингу) (рис. 1.7).

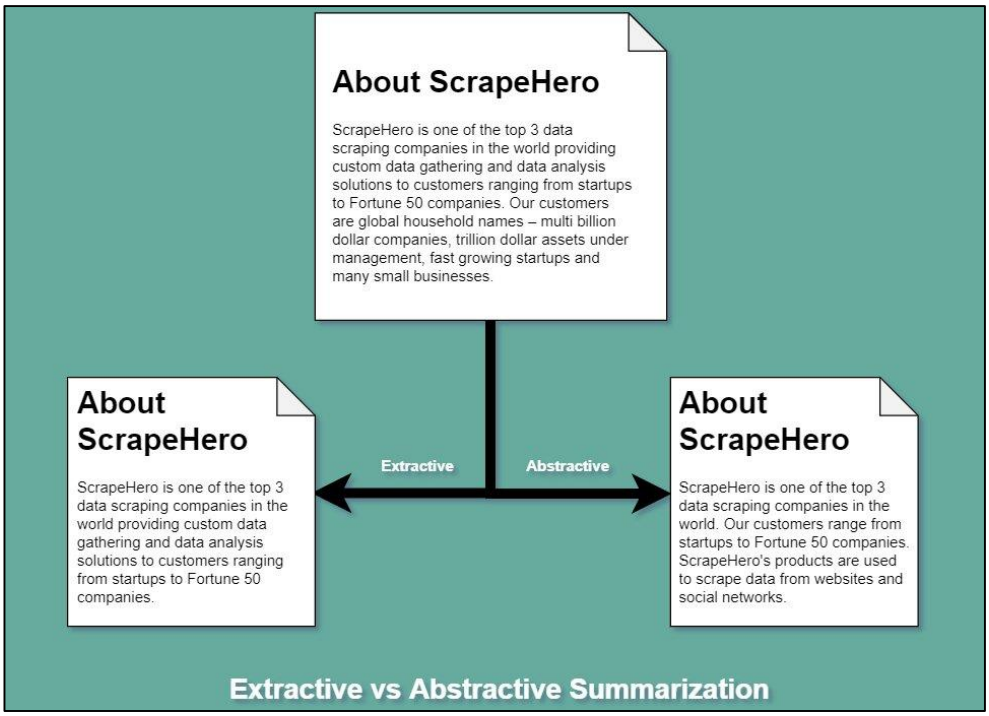

Рисунок 1.7 – Методи суммарайзингу (анотування) [7]

Екстракційний – методи екстракційного підсумовування працюють саме так. Він бере текст, ранжує всі речення відповідно до розуміння та релевантності тексту та представляє вам найважливіші речення.

Цей метод не створює нових слів чи фраз, він просто бере вже існуючі слова та фрази та представляє лише їх. Ви можете уявити це як взяти сторінку тексту та позначити найважливіші речення за допомогою маркера.

Абстрактний – з іншого боку, абстрактне резюмування намагається вгадати значення всього тексту та представляє його вам.

Він створює слова та фрази, об’єднує їх у змістовний спосіб і разом з цим додає найважливіші факти, знайдені в тексті. Таким чином, методи абстрактного підсумовування є складнішими, ніж методи екстрактивного підсумовування, а також обчислювально дорожчі.

Типовим підходом є моделювання послідовності до послідовності, оскільки вхідні дані є послідовністю слів, а резюме також є послідовністю слів. В архітектурі кодер-декодер кодер використовує LSTM для надання вхідного представлення. Декодер також є LSTM, який генерує вихідну послідовність. Рівень уваги між кодером і декодером допомагає визначити найбільш відповідні слова для резюме (рис. 1.8) [8].

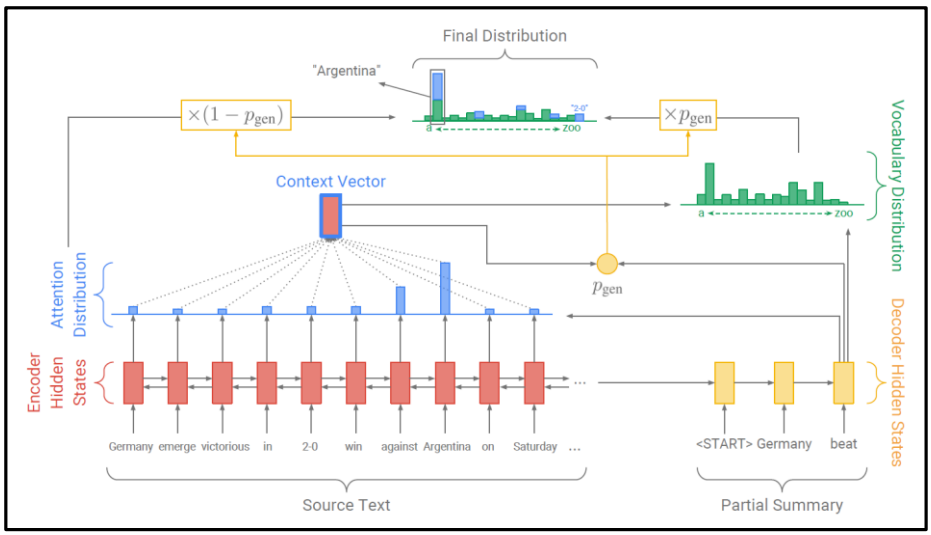

Рисунок 1.8 – Архітектура нейронної мережі для підсумовування тексту

Моделі Seq2seq, LSTM зробили можливим абстрактне підсумовування, навіть якщо вони ще не є найсучаснішими в порівнянні з методами екстрактивного підсумовування. Ці моделі навчаються наскрізно, не турбуючись про моделювання кожного кроку традиційного конвеєра підсумовування. Їм також не потрібен доступ до спеціальної лексики чи – попередня обробка. Цей наскрізний підхід був успішно застосований до коротких вихідних послідовностей, таких як заголовки новин або короткі відповіді електронною поштою.

У мережі генератора вказівників генератор надає нові слова, тоді як вказівник копіює слова з вихідного тексту. Моделі Seq2seq часто створюють повторювані речення. Модель покриття дозволяє уникнути повторів.

Фернандес та ін. показали, що кодери послідовності з графовим компонентом краще справляються зі фіксацією зв’язків на великій відстані.

Ось кілька повсякденних прикладів узагальнення тексту: заголовки новин, плани для студентів, попередні перегляди фільмів, протоколи зустрічей, біографії для резюме або некрологів, скорочені версії книг, випуск інформаційних бюлетенів, фінансові дослідження, патентні дослідження, аналіз юридичних контрактів, твіти про новий вміст, чат-боти, які відповідають на запитання, підсумки електронною поштою тощо [8].

Коли Пошук Google надає результати пошуку, деякі записи супроводжуються автоматично згенерованими підсумками. Google може використовувати для цієї мети графік знань. Підхід Google до узагальнення здебільшого орієнтований на об’єкти. Резюмування поширюється на часові шкали та події щодо сутностей.

Загальний підхід ML полягає в тому, щоб розглядати резюмування тексту як проблему класифікації. Алгоритм навчається під наглядом на основі оригінального тексту, витягненого резюме та набору функцій. Алгоритм вчиться класифікувати речення як короткі речення або речення без підсумку.

Класифікатори можуть бути засновані на наївному методі Байєса, деревах рішень, SVM, HMM і CRF. Часто кожне речення класифікується незалежно від інших. Однак, оскільки HMM і CRF фіксують залежності, вони перевершують інші методи (рис. 1.9).

Name	Description
Position	1/sentence no.
Doc_First	Whether it is the first sentence of a document
Para_First	Whether it is the first sentence of a paragraph
Length	The number of words in a sentence
Quote	The number of quoted words in a sentence

Table 1. Types of surface features

Name	Description
Centroid_Uni	The sum of the weights of centroid uni-gram
Centroid_Bi	The sum of the weights of centroid bi-grams
SigTerm_Uni	The number of signature uni-grams
SigTerm_Bi	The number of signature bi-grams
FreqWord_Uni	The sum of the weights of frequent uni-grams
FreqWord_Bi	The sum of the weights of frequent bi-grams

Table 2. Types of content features

Name	Description
FirstRel_Doc	Similarity with the first sentence in the document
FirstRel_Para	Similarity with the first sentence in the paragraph
PageRankRel	PageRank value of the sentence based on the sentence map

Table 3. Types of relevance features

Рисунок 1.9 – Деякі функції, які використовуються класифікатором ML для підсумовування тексту [8]

Проблема контрольованих алгоритмів полягає у створенні мічених даних для навчання. Ця проблема є гіршою для MDS. У напівконтрольованому підході використовується невелика кількість позначених даних разом із значно більшою кількістю не позначених даних. Алгоритм навчається ітераційно, класифікуючи деякі немічені дані в кожній ітерації [8].

Модель «мішок слів» (MC, bag-of-words) – дає змогу презентувати текст як невпорядковану колекцію слів (якщо бути точним, то токенів) без урахування граматичних правил і порядку розміщення їх у конкретному розділі тексту [6].

Можна навести один з найпоширеніших прикладів NLP, достатньо відкрити Google і сказати: «Окей, гугл ...». І ось пошукова система обробляє необхідний запит.

Але ця функція була б недоступна, якби не можливість пристрою зрозуміти природну мову, якою говорять люди. Здатність машини обробляти сказане, структурувати отриману інформацію, визначати необхідну дію у відповідь і відповідати мовою, зрозумілою користувачеві, і є NLP або Natural Language Processing.

З одного боку, все досить просто. Однак мова людини значно відрізняється від мови пристрою, робота. Основна складність для розробників у тому, що машина сприймає все буквально. А природна мова дуже насичена і наповнена багатозначними словами, омонімами, часто прихованими підтекстами, які вловлює не кожна людина [6].

Які завдання може вирішити NLP?

Пишеться чи говориться, на виході завжди виходить текст. Саме він є об'єктом обробки NLP. Таким чином, операції, які можна робити з текстом, навчаються реалізовувати за допомогою машини. Тому коло завдань перед NLP досить широке:

– машинний переказ;
– перевірка письменності текстів;
– розпізнавання мови та пошук відповідей;

– голосове управління;
– суммаризація – пошук основних фактів і переказ змісту тексту;
– аналіз настрою;
– класифікація текстів;
– показ відповідної онлайн реклами (пошук подібного контексту);
– прогнозування.

І з кожним днем завдань стає дедалі більше. А за рахунок того, що NLP настільки багатозадачне та може обробляти значну кількість даних, застосування йому знаходять у багатьох сферах діяльності [9].

Суммаризація – пошук основних фактів і переказ змісту тексту. Це саме те, що потрібно для розроблення інформаційної технології синтезу тексту про водні ресурси.

У NLP-моделі підсумовування тексту (суммаризації) автоматично скорочують документи, подкасти, статті, тощо до найважливіших фрагментів, завдяки передовим дослідженням глибокого та машинного навчання.

Команди продуктів інтегрують API текстового резюме у свої інтелектуальні платформи, щоб створити інструменти підсумовування на основі ШІ для співбесід, юридичних документів тощо.

Методи суммарайзингу. Методи резюмування тексту поділяються на дві основні категорії: екстрактивні та абстрактні. Резюмування екстракційного тексту, коли модель «витягує» найважливіші речення з оригінального тексту, є більш традиційним методом. Витягнене резюме тексту не змінює мову оригіналу, яка використовується в тексті. Навпаки, абстрактне підсумовування тексту вимагає, щоб сама модель генерувала підсумки, які можуть містити або не містити слова та/або речення з оригінального тексту [10].

Seaborn — це бібліотека візуалізації даних Python на основі matplotlib . Він забезпечує інтерфейс високого рівня для малювання привабливої та інформативної статистичної графіки. Seaborn допомагає досліджувати та розуміти свої дані [11].

NumPy – це бібліотека мови Python, що додає підтримку великих багатовимірних масивів і матриць, разом з великою бібліотекою високорівневих математичних функцій. Швидкі та універсальні концепції векторизації, індексації та трансляції є фактичними стандартами масивних обчислень сьогодні. NumPy пропонує комплексні математичні функції, генератори випадкових чисел, процедури лінійної алгебри, перетворення Фур'є тощо. NumPy підтримує широкий спектр апаратного забезпечення та обчислювальних платформ і добре працює з розподіленими бібліотеками, бібліотеками GPU та розрідженими масивами. Ядром NumPy є добре оптимізований код C. Насолоджуйтеся гнучкістю Python зі швидкістю скомпільованого коду. Високорівневий синтаксис робить його доступним і продуктивним для програмістів із будь-яким рівнем підготовки та досвіду [12].

Pandas – це пакет Python, який забезпечує швидкі, гнучкі та виразні структури даних, розроблені для того, щоб зробити роботу з «реляційними» або «міченими» даними одночасно легкою та інтуїтивно зрозумілою. Він має на меті стати основним будівельним блоком високого рівня для практичного аналізу реальних даних у Python. Крім того, він має ширшу мету — стати найпотужнішим і найгнучкішим інструментом аналізу/маніпулювання даними з відкритим кодом, доступним будь-якою мовою. Він уже на шляху до цієї мети [13].

Matplotlib – це бібліотека двовимірної графіки для мови програмування Python, за допомогою якої можна створювати високоякісні малюнки різних форматів. Matplotlib робить легкі речі легкими, а складні – можливими. Можливе створення якісних сюжетів для публікації. Створення інтерактивних фігур, які можна масштабувати, панорамувати, оновлювати. Налаштування візуального стилю і макету. Експорт у багато форматів файлів. Вставка в JupyterLab і графічний інтерфейс користувача [14].

Summarizer – бібліотека, яка узагальнює інформацію у великих текстах для швидшого використання без втрати важливої інформації [15].

Scikit-Learn – бібліотека для машинного навчання і прогнозної аналітики. Вона містить ряд методів, що охоплюють: алгоритми класифікації та регресії, кластеризації, валідацію і вибір моделей [16].

Warnings – бібліотека, яка видає попереджувальні повідомлення, як правило, видаються в ситуаціях, коли корисно попередити користувача про певний стан програми, коли ця умова (як правило) не вимагає збільшення винятку та припинення програми [17].

1.3 Проблема анотування опису географічної прив'язки

Під час виконання моєї бакалаврської кваліфікаційної роботи було розроблено додаток, який за кодом басейну знаходить населені пункти в ньому та виводить з книги чи іншого джерела, де було згадування даного населеного пункту. Але щоб не виписувати, не шукати все що там є, можна буде просто взяти текст з додатку та за допомогою розробленої інтелектуальної технології знайти і виписати основну інформацію (провести анотування) з книги чи іншого джерела.

На сьогоднішній день є певна проблема анотування опису географічної прив'язки.

Для прискорення процесу виведення інформації та підвищення якості картографічних документів, бажано застосовувати растеризатори (растрові процесори, RIP – Raster Image Processor) – спеціальні програми або обладнання, яке більш якісно виконує обробку (перетворення) растрово-векторної інформації, що надходить з баз даних та програмного забезпечення ГІС, в растр необхідної роздільної здатності для конкретного пристрою виведення інформації [18, 19].

Визначення масивів поверхневих вод та масивів підземних вод є складовою аналізу характеристик району річкового басейну.

Згідно з Водним кодексом України масив поверхневих вод (МПВ) – це водний об’єкт (річка, озеро, та інше) або його частина. Окремий МПВ – окремий пункт моніторингу [20].

Важливим є опис такої прив’язки, тому важливо матеріал теж використовувати в анотації.

Зазвичай всі роботи з картами роблять в наступних пакетах: «ArcGIS Desktop», ГІС-пакет «GIS Map Viewer», ГІС-пакет «Digitals», QGIS.

ArcGIS Desktop – американська лінія ГІС-продуктів, яка призначена для настільних комп'ютерів. Вона включає в себе ArcView, ArcEditor та ArcInfo. В свою чергу ArcView поділяється на ArcMap, ArcCatalog та ArcToolbox [21].

Одним з недоліків системи є ціна даної програми, але в програми є безкоштовна версія, фунціонал якої обмежений. Вигляд додатку зображено на рисунку 1.10.

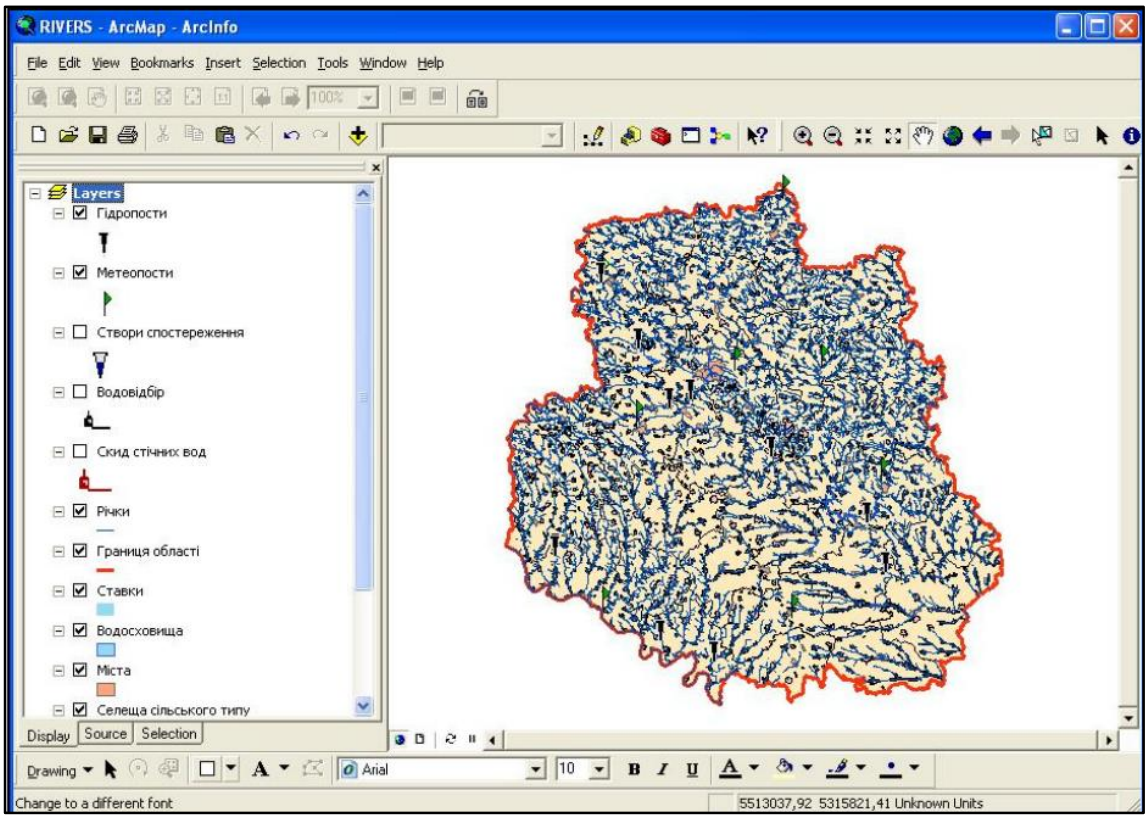

Рисунок 1.10 – Вигляд головного вікна ArcGIS

GIS Map Viewer – це універсальна геоінформаційна система, що має засоби створення та редагування цифрових карт і планів міст, обробки даних ДЗЗ, виконання різних вимірів і розрахунків, оверлейних операцій, побудови 3D моделей, обробки растрових даних, засоби підготовки графічних документів в цифровому та друкованому вигляді, а також інструментальні засоби для роботи з базами даних [22].

Вигляд додатку зображено на рисунку 1.11.

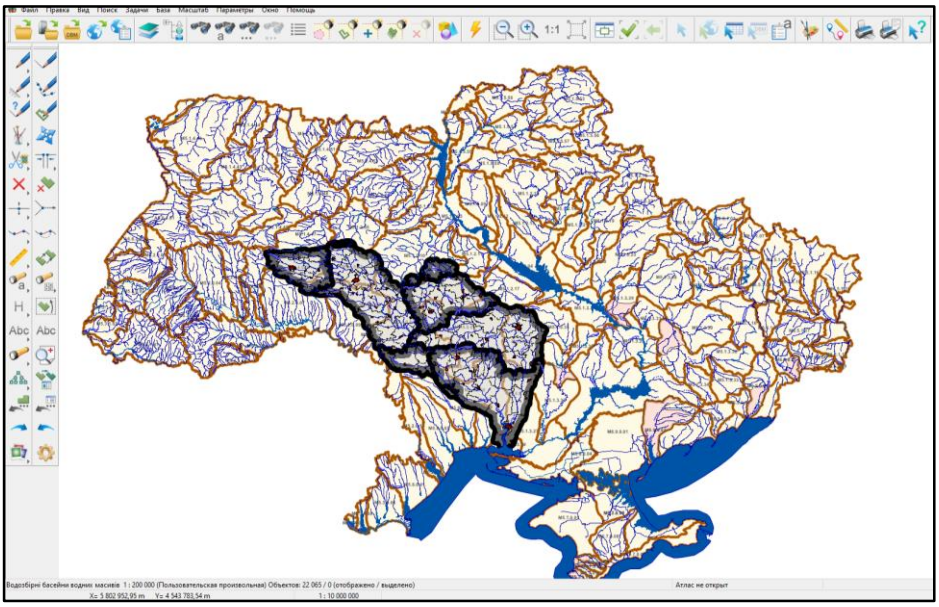

Рисунок 1.11 – Вигляд головного вікна GIS Map Viewer

Digitals – це програмне забезпечення від українського виробника - компанії "Геосистема" (м. Вінниця), призначене для вирішення завдань цифрової картографії й землевпорядкування. Дана система є в двох варіантах: Standard та Professional [23].

Вигляд додатку зображено на рисунку 1.12.

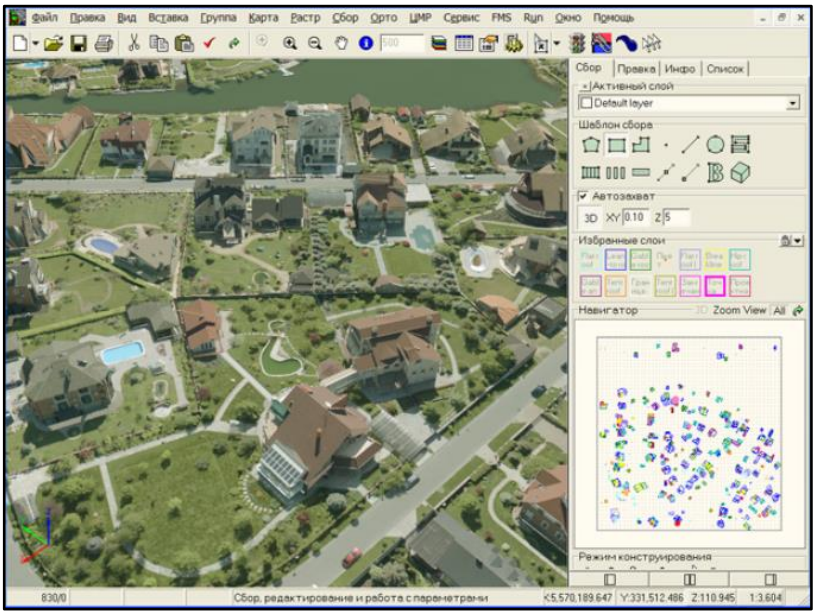

Рисунок 1.12 – Вигляд головного вікна ГІС «Digitals»

Також під час роботи з картами часто використовується матриця висот (рис. 1.13).

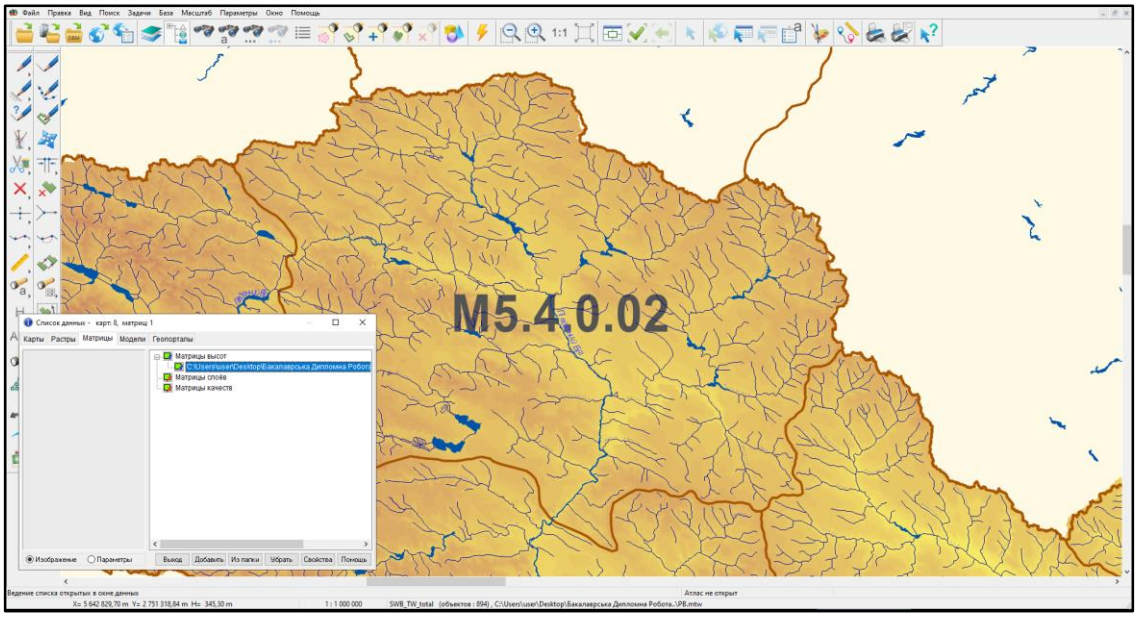

Рисунок 1.13 – Матриця висот

Для прикладу розглянемо сайт «Visicom» [24]. В ньому є можливість пошуку місцевості за назвою і текстової інформації про нього, створення картографічних продуктів для ГІС-проектів. Використання регіональних

моделей країн, 2D, 2.5D, 3D карт, моделей рельєфу і багато іншого. Сайт надає безкоштовну API з можливістю до 1 500 запитів безкоштовно. Також сайт дозволяє отримати географічні координати об’єкта по його опису і навпаки, тобто певна прив’язка також тут є.

Розглянемо і цей сайт. На даному сайті є місце для пошуку, куди користувач вводить дані, які потрібно знайти (рис 1.14).

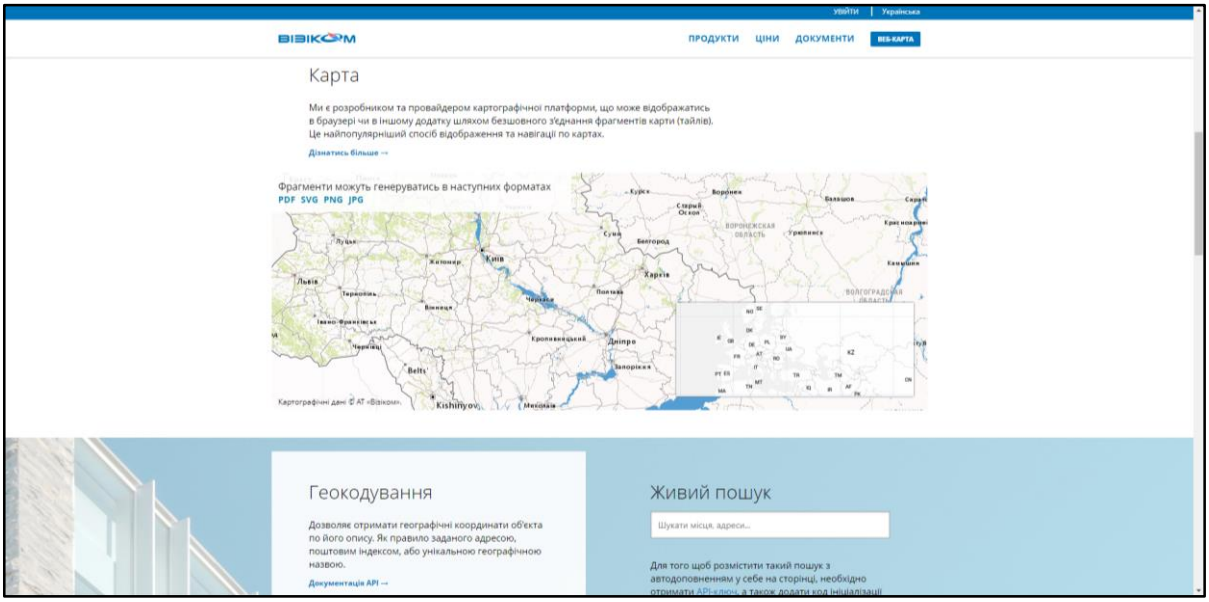

Рисунок 1.14 – Головна сторінка сайту

Вводимо назву населеного пункту і система пропонує різні уточнюючі варіанти місцевості, інформацію/місцезнаходження якого хочемо знайти (рис. 1.15).

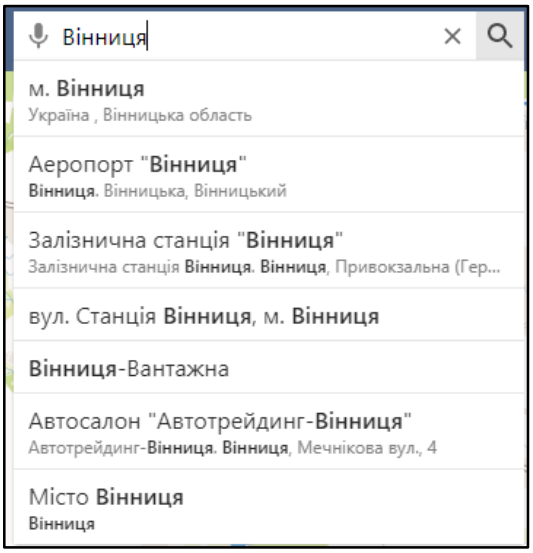

Рисунок 1.15 – Пошуковий запит

Після цього система покаже на карті місцевість, яку потрібно було знайти, але без різної інформації про неї (рис. 1.16).

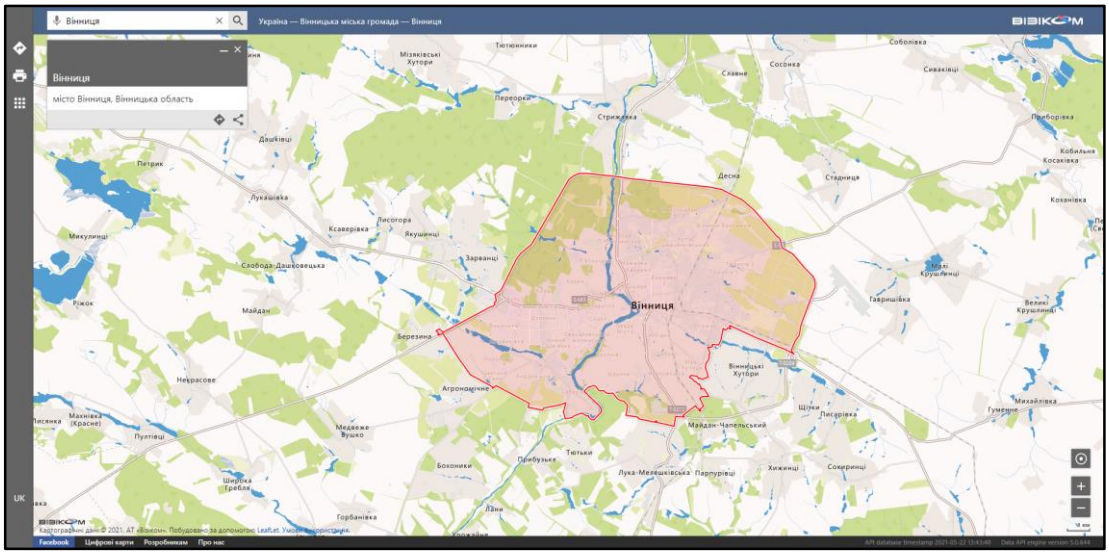

Рисунок 1.16 – Результат пошуку

А завдяки тому, що використовувались ГІС-технології, а саме – розподіл частини території Вінницької області по басейнах, можна отримати інформацію про водні ресурси певного міста, залежно від того, по якому PDF-файлу буде здійснюватися пошук. Розглянемо це на прикладі.

Є файл з назвами річок, водосховищ та населених пунктів, які до них відносяться, українською та англійською мовами відповідно, код їх басейнів та тип (рис. 1.17).

Name of WB (UA)	Name of WB (EN)	Code of WB	Type of WB (UA)	Type of WB (EN)	UA-name entities	EN-name entities
1	2	3	4	5	6	7
Південний Буг	Pivdenny Bug	UA_M5.4_0010	річка	river	с. Лисогірка, с. Педоси, с. Лозова, с. Думенки, с. Нова Синявка, с. Чудинці, с. Березна	Lysohirka, Pedosy, Lozova, Dumenky, Nova Syniavka, Chydunovtsi, Berezna
Південний Буг	Pivdenny Bug	UA_M5.4_0012	річка	river	с. Кам'яногірка, с. Пиківська Слобідка, с. Матяшівка, с. Іванопіль, с. Слобідка, с. Олександрівка, с. Курилівка, с. Уладівка, с. Журавне, с. Широка Гребля, с. Порик, с. Томашпіль	Kamianohirka, Pykivs'ka Slobidka, Matiashivka, Slobidka, Oleksandrivka, Kurylivka, Uladivka, Zhuravne, Shyroka Hrebla, Poryk, Tomashpil
Тесівка	Tesivka	UA_M5.4_0078	річка	river	с. Кусиківці, с. Медведівка, с. Шевченка, с. Іванівці, с. Теси, с. Лисогірка	Kusykivtsi, Medvedivka, Shevchenka, Ivanivtsi, Tesy, Lysohirka
Тесівка	Tesivka	UA_M5.4_0079	річка	river	с. Осінок, с. Теси, с. Новостянтинів	Osinok, Tesy, Novokostiantyniv
Без назви	No name	UA_M5.4_0080	річка	river	с. Гавришівка, с. Майдан-Вербецький, с. Бохни, с. Ялинівка, с. Ліско-Березівка, с. Новомиколаївка, с. Розсохувата, с. Москалівка, с. Лука	Havryshivka, Maidan,Verbets'kyi, Bokhny, Yalynivka, Lisо-Berezivka, Novomykolaivka, Rozsokhuvata, Moskalivka, Luka
Іква	Ikva	UA_M5.4_0087	річка	river		
Пожарка	Pozharka	UA_M5.4_0092	річка	river	с. Подоляни, с. Залісся, с. Ольгине, с. Паплинці, с. Адампіль, с. Гончариха, с. Терешпіль, с. Перекора	Podoliany, Zalissia, Olhyne, Paplyntsi, Adampil, Hocharykha, Tereshpil, Perekora
Пожарка	Pozharka	UA_M5.4_0093	річка	river	с. Крупин, с. Куманівці, с. Залісся	Krupyn, Kumanivtsi, Zalissia
Пожарка	Pozharka	UA_M5.4_0094	річка	river	с. Березна	Berezna
Пожарка	Pozharka	UA_M5.4_0095	річка	river	с. Березна	Berezna
Велика Руда	Velika Ruda	UA_M5.4_0096	річка	river	с. Малий Митник, с. Великий Митник, с. Жданівка, с. Ольгине, с. Кушелівка, с. Качанівка, с. Українське, с. Семки, с. Терешпіль, с. Софіївка, с. Держанівка	Malyi Mytnyk, Velykyi Mytnyk, Zhdanivka, Olhyne, Kushelivka, Kachanivka, Ukrains'ke, Semky, Tereshpil, Sofiivka, Derzhanivka
Велика Руда	Velika Ruda	UA_M5.4_0097	річка	river	м. Хмільник, с. Будків, с. Філіопіль, с. Малий Митник, с. Великий Митник, с. Дібрівка	Khmil'nyk, Budkiv, Filiopil, Malyi Mytnyk, Velykyi Mytnyk, Dibrivka
Снивода	Snivoda	UA_M5.4_0098	річка	river	с. Семки, с. Митинці, с. Торчин, с. Держанівка, с. Сулківка, с. Скаржинці, с. Мар'янівка, с. Лисогірка	Semky, Mytyntsi, Torchyn, Derzhanivka, Sulkivka, Skarzhyntsi, Marianivka, Lisohirka
Снивода	Snivoda	UA_M5.4_0099	річка	river	с. Маркуші, с. Рибчинці, с. Уланів, с. Сальниця	Markushi, Rybchyntsi, Ulaniv, Sal'nytsia
Снивода	Snivoda	UA_M5.4_0101	річка	river	с. Пролетар, с. Лозна, с. Воронівці	Proletar, Lozna, Voronivtsi
Снивода	Snivoda	UA_M5.4_0103	річка	river	с. Шепівка, с. Кривоший, с. Хутори-Кривошиїнецькі	Shepivka, Kryvoshyi, Khutory-Kryvoshyinets'ki
Снивода	Snivoda	UA_M5.4_0105	річка	river	с. Пиків	Pykiv
Снивода	Snivoda	UA_M5.4_0107	річка	river	с. Іванів, с. Слобідка, с. Жигалівка, с. Яблуневе	Ivaniv, Slobidka, Zhyhalivka, Yabluneve
Сальницька	Salnitska	UA_M5.4_0108	річка	river	с. Уланів, с. Морозівка, с. Сальниця, с. Лип'ятин, с. Лісогірка, с. Гнатівка	Ulaniv, Morozivka, Sal'nytsia, Lypiatyn, Lisohirka, Hnativka
Витхла	Whithla	UA_M5.4_0109	річка	river	с. Кустовецьке, с. Малий Острожок, с. Крижанівка, с. Великий Острожок, с. Слобода-Кустовецька, с. Вишенька, с. Червоний Степ, с. Ступник, с. Кропивна, с. Клітенка, с. Подорожня, с. Польова Слобідка	Kustovets'ke, Malyi Ostrozchok, Kryzhanivka, Velykyi Ostrozchok, Slovoda-Kustovets'ka, Vyshen'ka, Chervonyi Step, Stupnyk, Kropyvna, Klitenka, Podorozhnia, Polova Slobidka
Витхла	Whithla	UA_M5.4_0110	річка	river	с. Пагурці, с. Тараски, с. Великий Острожок, с. Чепелі, с. Петриківці, с. Смила	Pahurtsi, Tarasky, Velykyi Ostrozhok, Chepeli, Petrykivtsi, Smila

Рисунок 1.17 – Файл з назвами населених пунктів

Також є PDF-файл з інформацією про річку Південний Буг з назвами міст в ньому (рис. 1.18).

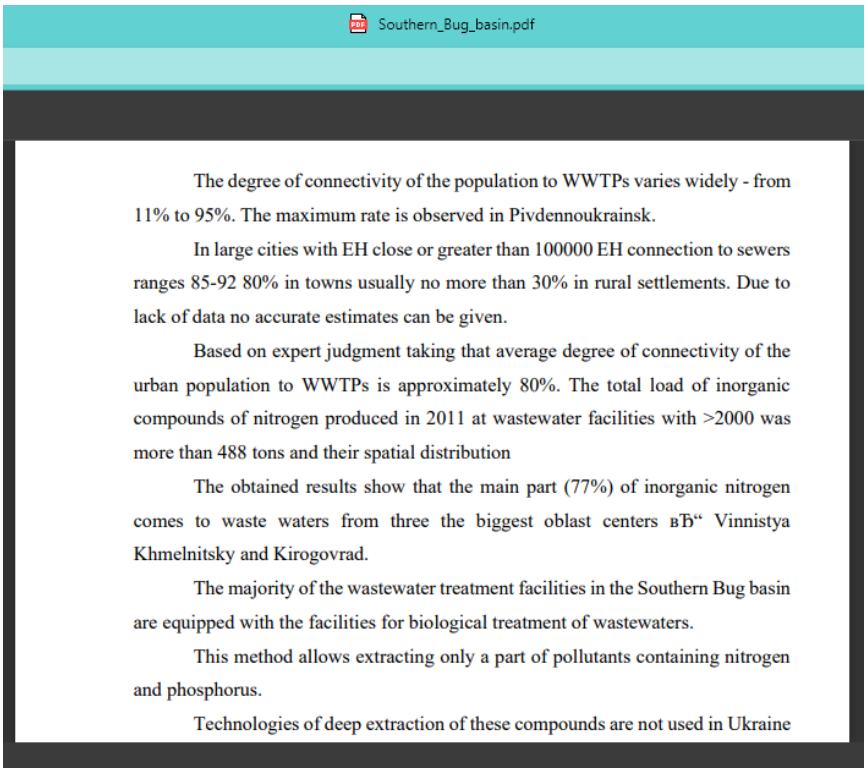

Рисунок 1.18 – PDF-файл з текстом для пошуку

Для початку оберемо MS Excel файл з нашими даними про населені пункти, та PDF-файл, в якому буде здійснюватися пошук (рис. 1.19 – 1.21).

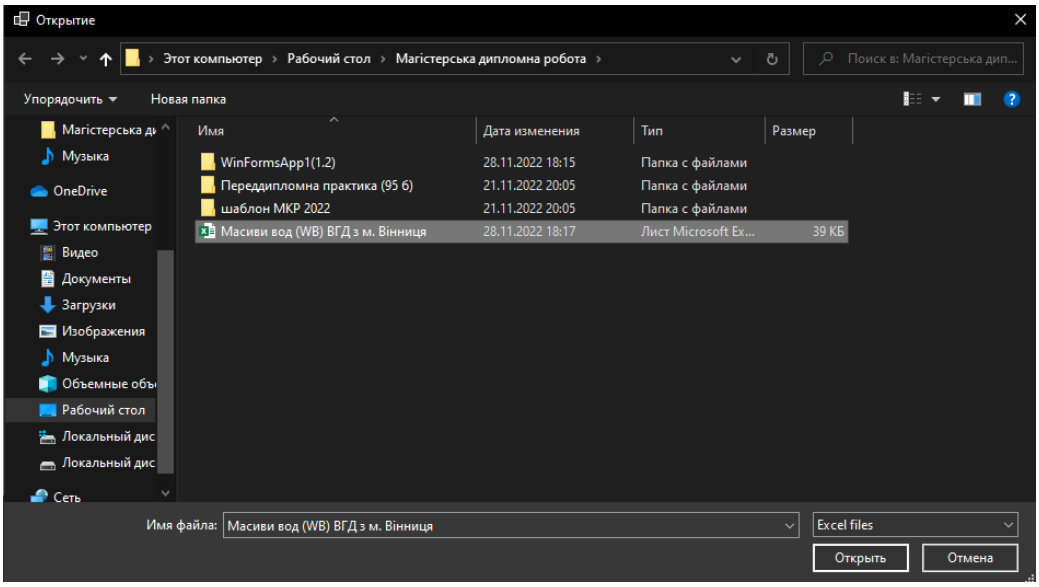

Рисунок 1.19 – Вибір MS Excel файлу

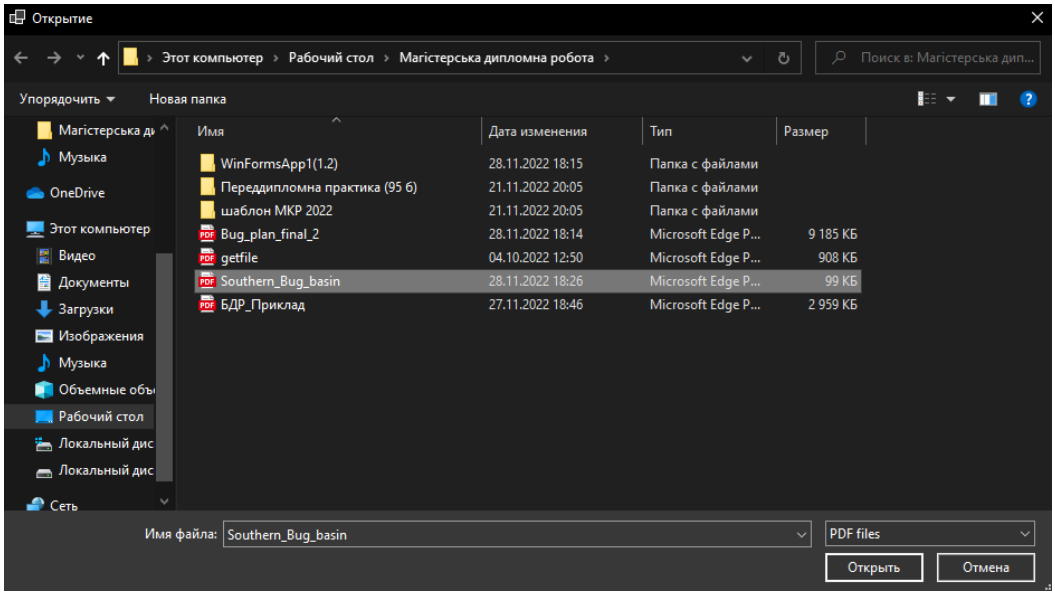

Рисунок 1.20 – Вибір PDF-файлу

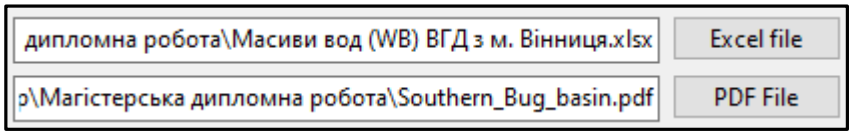

Рисунок 1.21 – Вибір файлів для роботи

Далі введемо код масиву вод, по якому з MS Excel файлу витягуються дані про населені пункти (рис. 1.22).

Рисунок 1.22 – Введення коду масиву вод

Відповідно до коду масиву вод додаток покаже 2 списки з назвами населених пунктів на українській та англійській мовах та перевіримо згідно файлу чи воно виводить те що потрібно (рис. 1.23, 1.24).

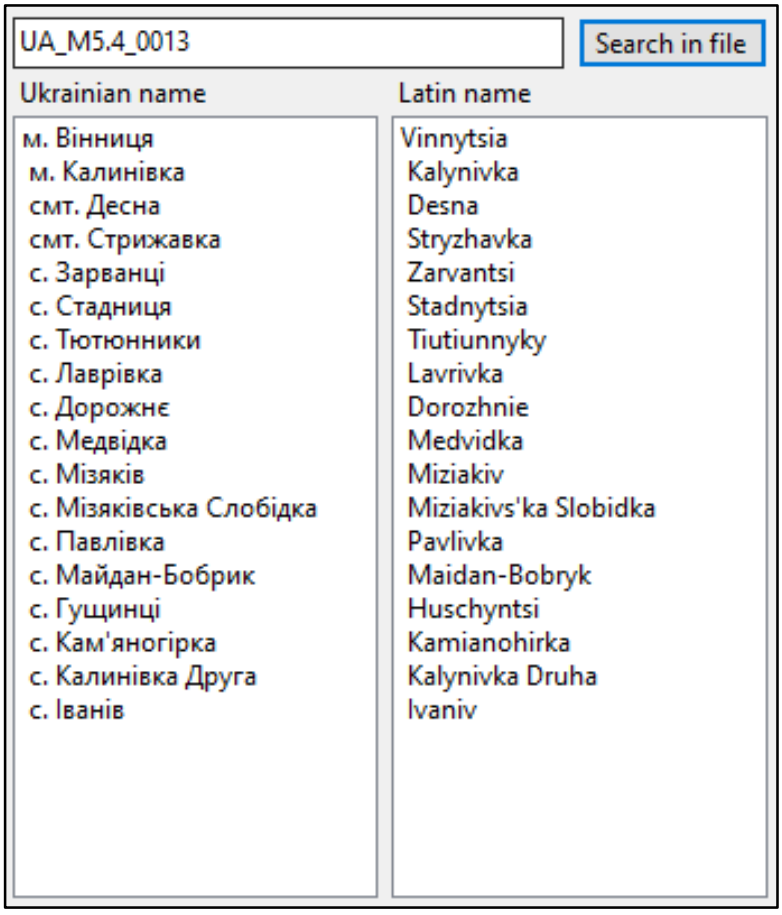

Рисунок 1.23 – Показ назв населених пунктів

UA_M5.4_0013	водосховище	reservoir	м. Вінниця, м. Калинівка, смт. Десна, смт. Стрижавка, с. Зарванці, с. Стадниця, с. Тютюнники, с. Лаврівка, с. Дорожнє, с. Медвідка, с. Мізяків, с. Мізяківська Слобідка, с. Павлівка, с. Майдан-Бобрик, с. Гущинці, с. Кам'яногірка, с. Калинівка Друга, с. Іванів	Vinnytsia, Kalynivka, Desna, Stryzhavka, Zarvantsi, Stadnytsia, Tiutiunnyky, Lavrivka, Dorozhnie, Medvidka, Miziakiv, Miziakivs'ka Slobidka, Pavlivka, Maidan-Bobryk, Huschyntsi, Kamianohirka, Kalynivka Druha, Ivaniv

Рисунок 1.24 – Перевірка

Тепер можна обрати населений пункт та натиснути кнопку «Search by …» того списку, з якого потрібно знайти дані, в даному випадку це англомовна (латинська) назва (рис. 1.25).

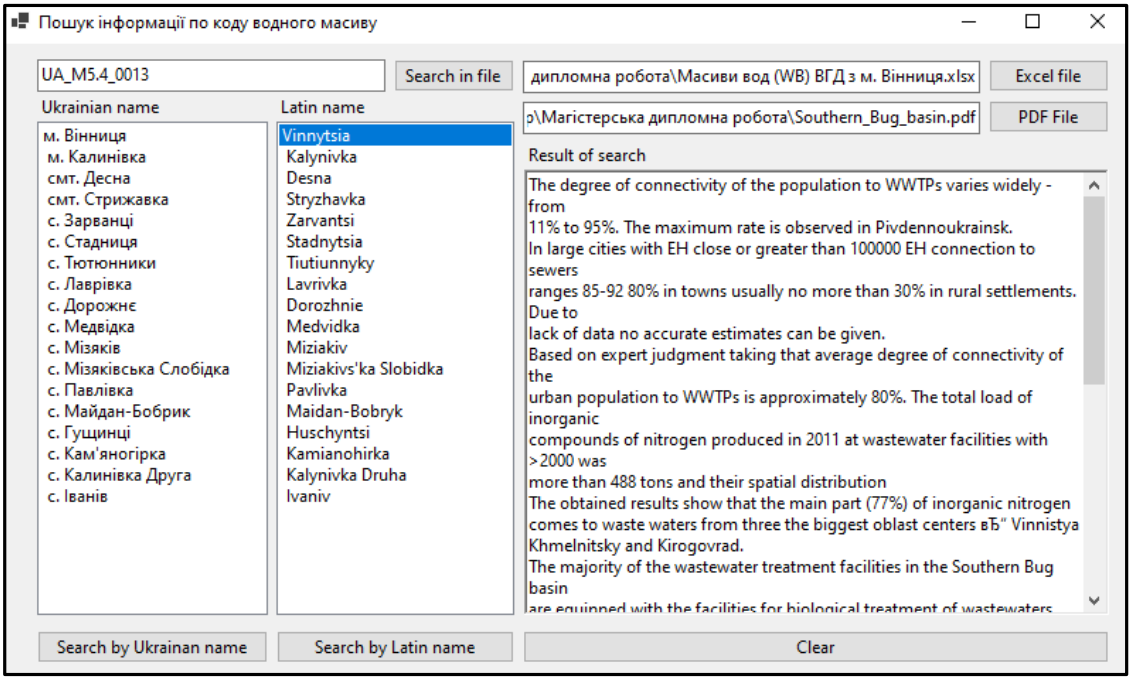

Рисунок 1.25 – Результат пошуку

Як можна побачити, результатом пошуку є дана інформація про певний населений пункт.

Давайте відкриємо сам PDF-файл та перевіримо чи дійсно цей текст з даного файлу (рис 1.26).

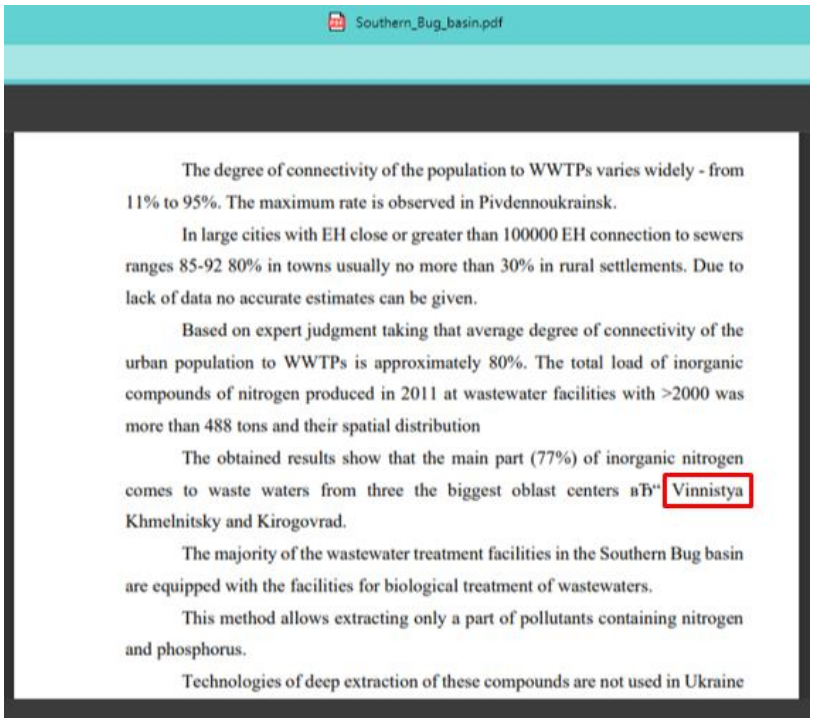

Рисунок 1.26 – Сторінка з PDF-файлу

В наступному розділі проведемо анотування та переглянемо чи гарний результат показує код з даним файлом.

1.4 Висновки

У даному розділі визначено суть наявної проблеми, розглянуто існуючі методи вирішення даної проблеми, аналог, а саме – було проведено тестування синтезу тексту методами штучного інтелекту, розглянуто бібліотеки, які будуть використовуватись у роботі, розглянуто проблему анотування тексту в ГІС-технологіях та їх геоприв’язування чи геокодування.

2 ОСНОВНІ ЕТАПИ ВИКОНАННЯ РОБОТИ ТА АВТОМАТИЧНІ ЕКОЛОГІЧНІ ЗВІТИ І КЛАСИФІКАЦІЯ НОВИН

2.1 Огляд датасету

Кожну секунду у відкритому доступі з'являється нова інформація про навколишнє середовище: публікуються звіти, книги, статті, новини тощо різними мовами. Автоматичне анотування дозволить більш ефективно її обробляти та використовувати для прийняття рішень.

Для роботи використовується датасет «NLP : Reports & News Classification» ENG & UKR Automatic Environmental Reports & News Classification [25].

Датасет взято з англійського видання книги «River Basin Management Plan for Pivdenny Bug: river basin analysis and measures» [26].

Джерело інформації було видано у друкарні англійською та українською мовами двома окремими книгами. Співавтор обох видань цієї книги є співавтором цього набору даних.

1. Речення скопійовані з книги (англ. – з англомовного видання, укр. – з українського видання) у форматі PDF разом із різними спец символами (номерами сторінок тощо), щоб був певний «шум».

2. Деякі спеціальні символи були вибірково видалені, включаючи деякі коми та крапки.

3. Визначення цільових ознак виконано авторами набору даних.

4. Бази даних англійською та українською мовами містять приблизно 95% однакової інформації.

Як правило, усі датасети з даними про синтез тексту методами штучного містять базові ознаки [27]. В нашому датасеті текст про забруднення річки Південний Буг, ознаки показані на рисунку 2.1.

– «text» – одне або кілька речень із звітів чи новин;
– «env_problems» – чи йдеться у тексті про екологічну проблему;

– «pollution» – чи текст про забруднення навколишнього середовища;

– «treatment» – текст про очисні споруди чи екологічні технології;

– «climate» – чи текст про кліматичні показники;

– «biomonitoring» – текст про біологічний, біотичний моніторинг у воді чи в басейні річки.

	text	env_problems	pollution	treatment	climate	biomonitoring
0	In the Southern Bug basin wastewater is discha...	1	1	0	0	0
1	This amount is distributed as follows: urban a...	1	1	0	0	0
2	Despite the similar volumes of discharged wast...	1	1	0	0	0
3	They bring 84% of organic pollution 86% of pho...	1	1	0	0	0
4	The input of the industry is between 7–21% and...	1	1	0	0	0

Рисунок 2.1 – Ознаки датасету « NLP : Reports & News Classification »

2.2 Розвідувальний аналіз

Проведемо розвідувальний аналіз даних по критерію забруднення (pollution).

Переглянемо середню довжину слова в тексті (рис. 2.2).

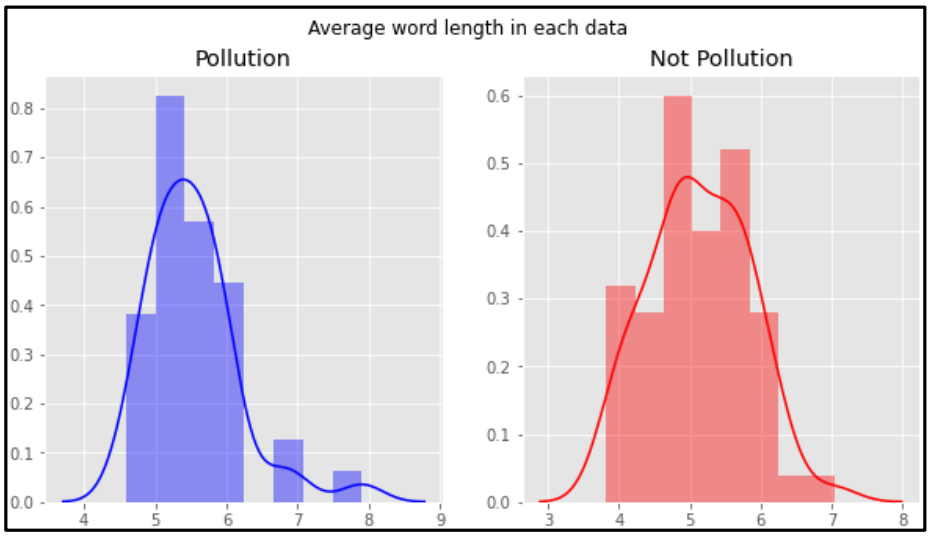

Рисунок 2.2 – Середня довжина слів

Найбільша кількість слів – з 5 та 6 символами.

Також переглянемо загальні стоп-слова в тексті та їх кількість в «Pollution» та «Not pollution» даних (рис 2.3, 2.4).

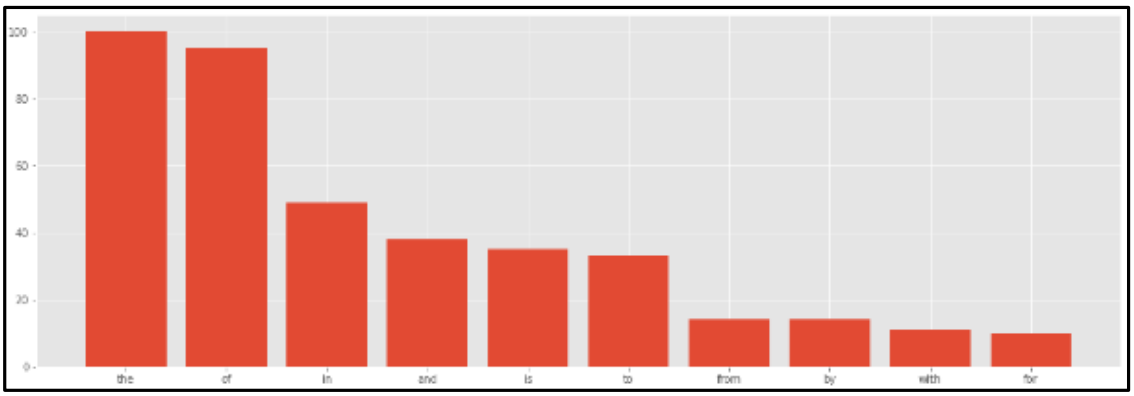

Рисунок 2.3 – Кількість стоп-слів в «Pollution» даних

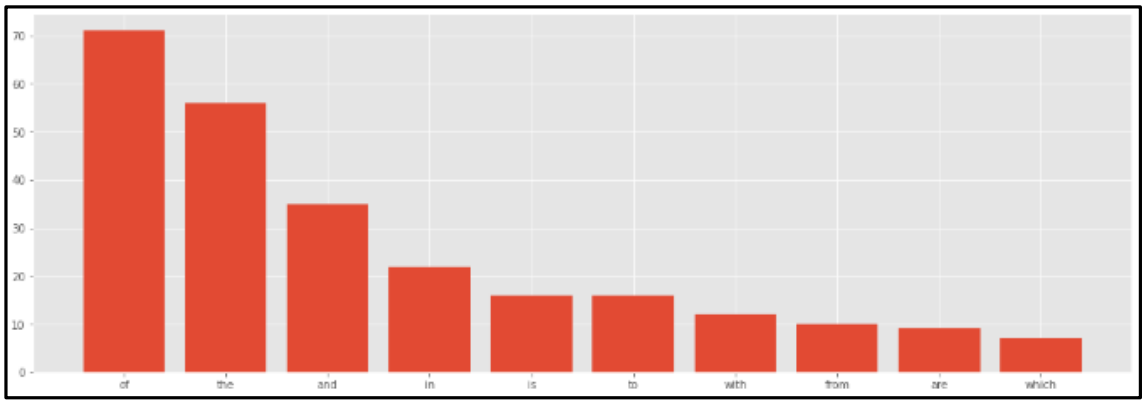

Рисунок 2.4 – Кількість стоп-слів в «Not Pollution» даних

Серед стоп-слів найпоширенішими є «the» та «of».

Також доволі цікавим буде переглянути загальновживані слова (рис. 2.5).

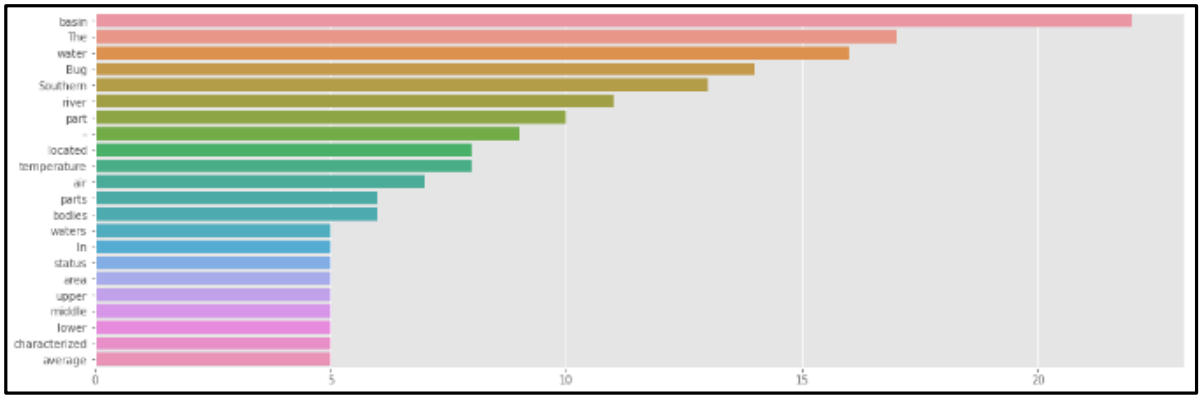

Рисунок 2.5 – Загальновживані слова

Найбільш загальновживаним є слово «basin» - басейн, що не є дивним, тому що текст про річку «Південний Буг». Також є проблемою що одним з найпоширеніших є слово «The», тому цей текст потребує очищення.

Далі йде створення хмари слів із значеннями під кадром даних категорії, загальновживані в обох типах (pollution and not pollution) (рис. 2.6, 2.7).

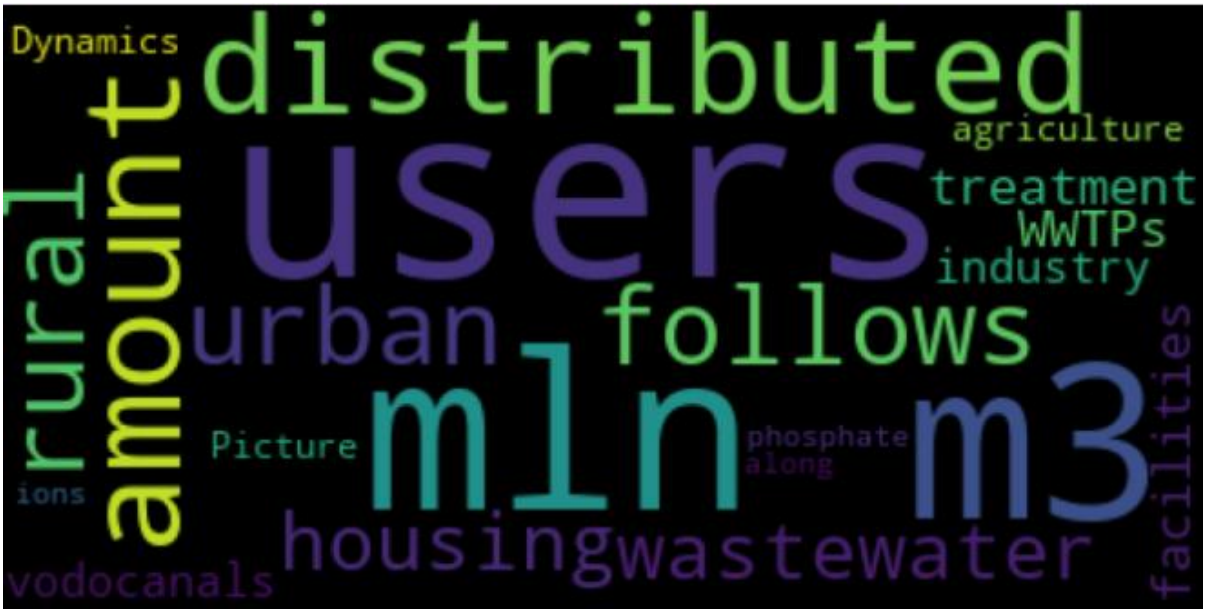

Рисунок 2.6 – Хмара слів

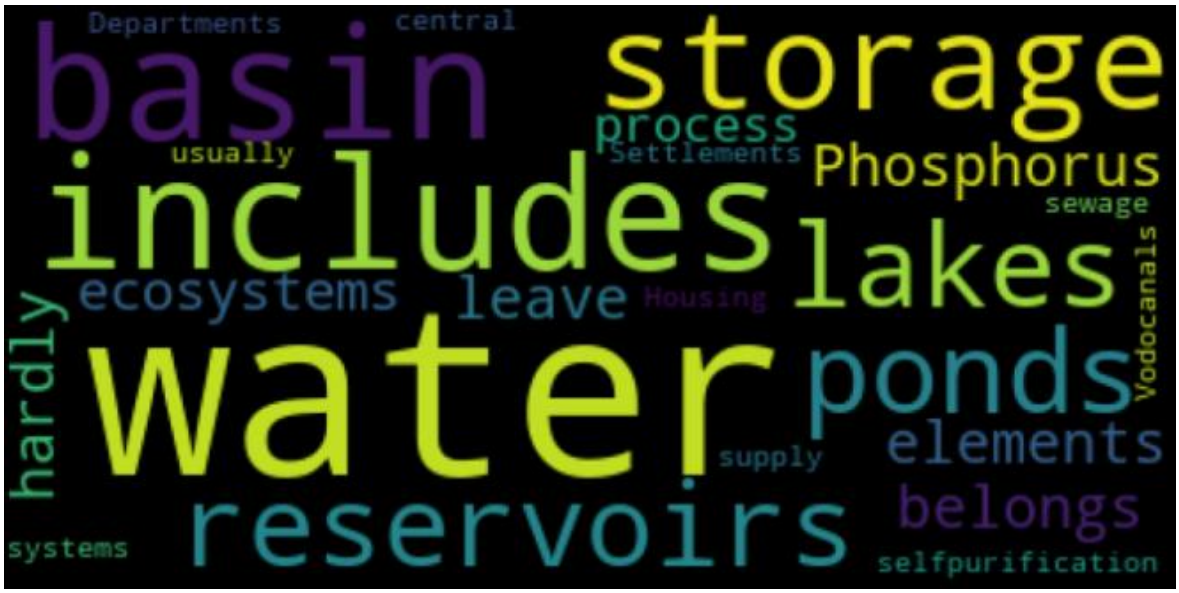

Рисунок 2.7 – Хмара слів

По даним «хмарам слів» можна помітити що одними з найбільш вживаних слів є «басейн», «вода», «резервуар», «забруднення» та інші.

Проведемо візуалізацію вкладень (рис. 2.8).

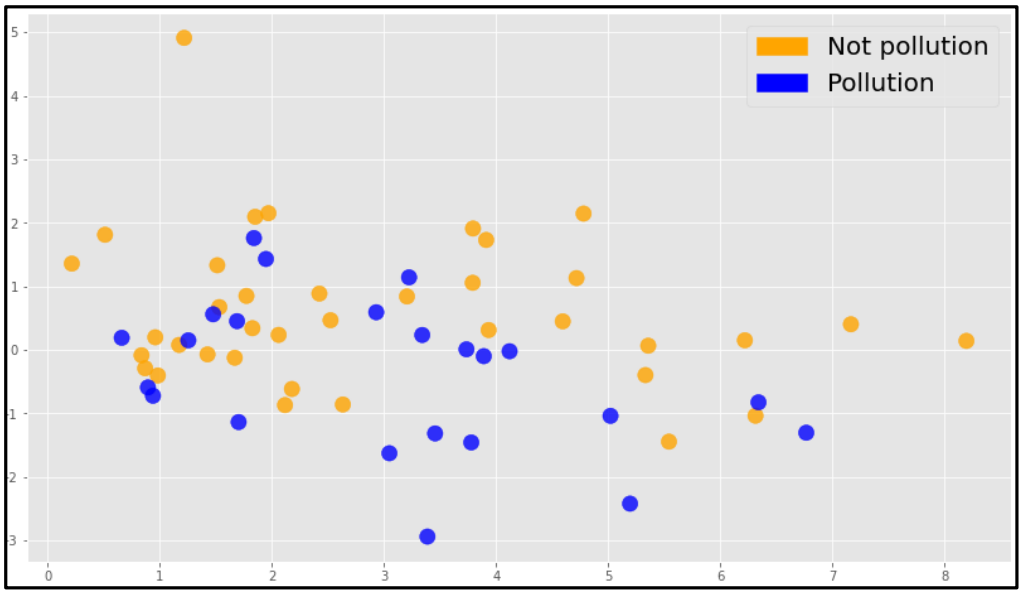

Рисунок 2.8 – Візуалізація вкладень

Ці вкладень виглядаються не дуже чітко розділеними. Перевіримо чи можна їх розділити.

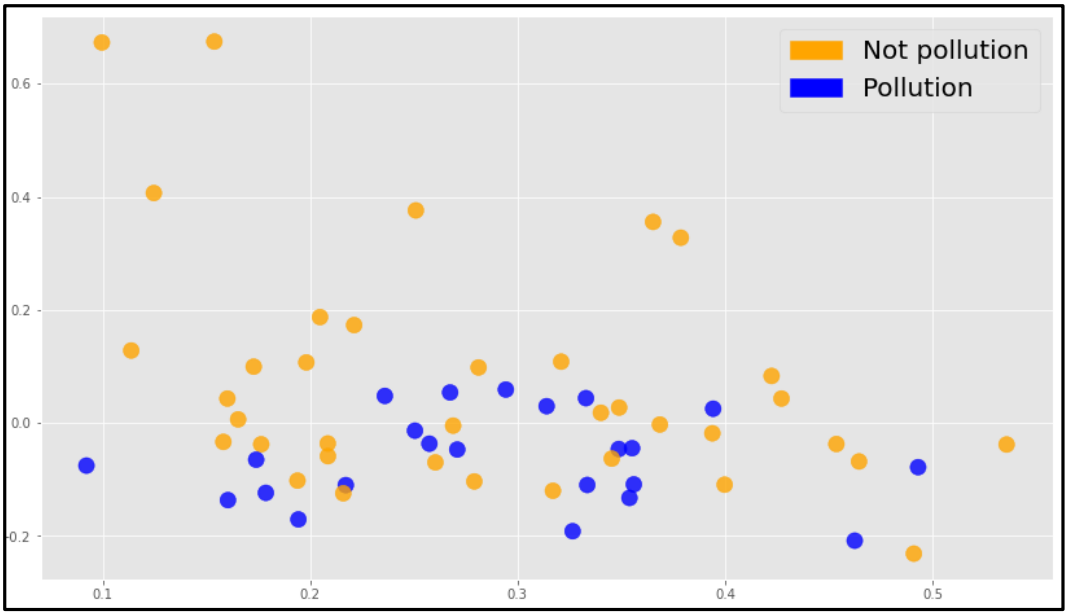

Рисунок 2.9 – Встановлення корисної моделі

2.3 Вибір оптимальних технологій

Суммарайзинг, тобто зменшення розміру тексту зі збереженням сенсу є однією з найбільш досліджуваних областей нейролінгвістичного програмування (NLP).

Техніки суммарайзингу, залежно від того, чи розглядаються точні речення так, як вони з'являються в оригінальному тексті, чи нові речення генеруються за допомогою методів обробки природної мови, поділяються на екстрактивні та абстрактні техніки. Екстракційне конспектування було дуже широко дослідженою темою, яка досягла стадії зрілості. Зараз дослідження змістилися в бік абстрактного узагальнення. Складність, що лежить в основі тексту природною мовою, робить абстрактне реферування важким і складним завданням [28].

Завдяки великій кількості даних, доступних сьогодні, підсумовування (суммарайзинг) тексту стало дуже важливим для отримання потрібної кількості інформації з величезних текстів. Ми бачимо довгі статті на веб-сайтах новин, у блогах, на веб-сайтах із відгуками клієнтів тощо. Для різних

методів, які досі використовувалися для підсумовування тексту, вивчалися різні статті [29].

Застосування обробки природної мови (NLP).

Ось кілька застосувань NLP, які використовуються в нашому повсякденному житті:

- автовиправлення та автозаповнення пошуку;
- мовний переклад;
- класифікація тексту;
- аналіз настроїв;
- перевірка орфографії;
- розпізнавання мови;
- моніторинг соціальних мереж.

Є декілька популярних NLP-моделей, які можна використати в даній роботі. Наприклад: BERT, PaLM, GPT-2, GPT-3, XLNet, T5.

BERT (Bidirectional Encoder Representations від Transformers) — на відміну від останніх моделей представлення мови, BERT розроблено для попереднього навчання глибоких двонаправлених представлень шляхом спільного обумовлення як лівого, так і правого контексту на всіх рівнях. У результаті попередньо підготовлені представлення BERT можна налаштувати лише з одним додатковим рівнем виводу для створення найсучасніших моделей для широкого кола завдань, таких як відповіді на запитання та мовні висновки, без суттєвих конкретних завдань модифікації архітектури.

Серед завдань класифікації BERT використовувався для класифікації фейкових новин і класифікації пар речень.

BERT все ще – нова, і в майбутньому може з’явитися багато нових застосувань. BERT можна використовувати для кількісної торгівлі. BERT можна застосовувати до певних доменів, але нам знадобляться попередньо навчені моделі для конкретного домену [30].

BERT надихнув на створення багатьох варіантів: RoBERTa, XLNet, MT-DNN, SpanBERT, VisualBERT, K-BERT, HUBERT тощо. Деякі варіанти

намагаються стиснути модель: TinyBERT, ALERT, DistilBERT тощо (рис. 2.10).

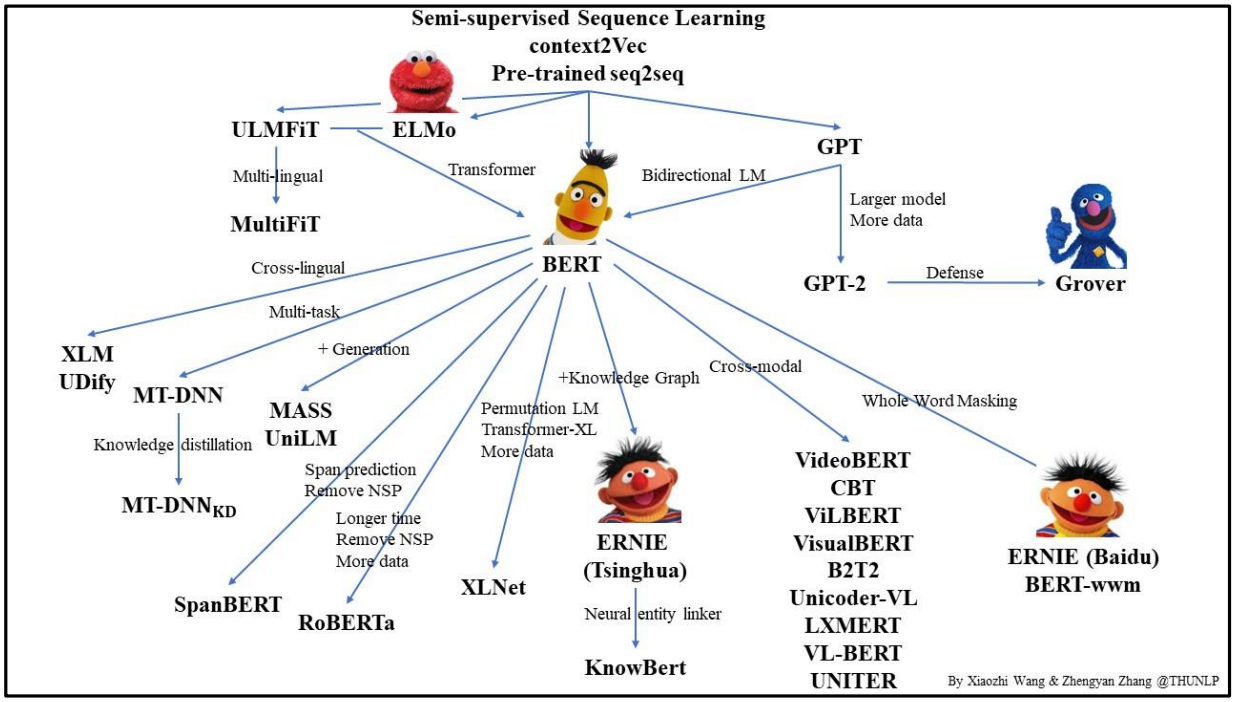

Рисунок 2.10 – Варіанти BERT [30]

Попередньо навчені моделі BERT доступні в двох розмірах:

База: 12 шарів, 768 прихованих розмірів, 12 головок самоуважності, 110M параметрів.

Великий: 24 шари, 1024 прихованих розмірів, 16 головок самоуважності, 340M параметрів.

Для кожного з перерахованих вище знадобилося 4 дні для навчання на 4 хмарних TPU (базовий) або 16 хмарних TPU (великий).

Для попереднього навчання використовувався розмір партії з 256 послідовностей. Кожна послідовність містила 512 токенів, тобто 128 тисяч токенів на партію. Корпус для попереднього навчання BERT містив 3,3 мільярда слів: 800 мільйонів з BooksCorpus і 2500 мільйонів з Wikipedia [35].

BERT є концептуально простим і емпірично потужним. Він отримує нові найсучасніші результати для одинадцяти завдань обробки природної мови, включаючи підвищення тесту GLUE до 80,4% (7,6% абсолютне покращення),

точність MultiNLI до 86,7 (5,6% абсолютне покращення) і питання SQuAD v1.1 відповідаючи на тест F1 до 93,2 (1,5% абсолютного покращення), перевершуючи показники людини на 2,0% (рис. 2.11) [31].

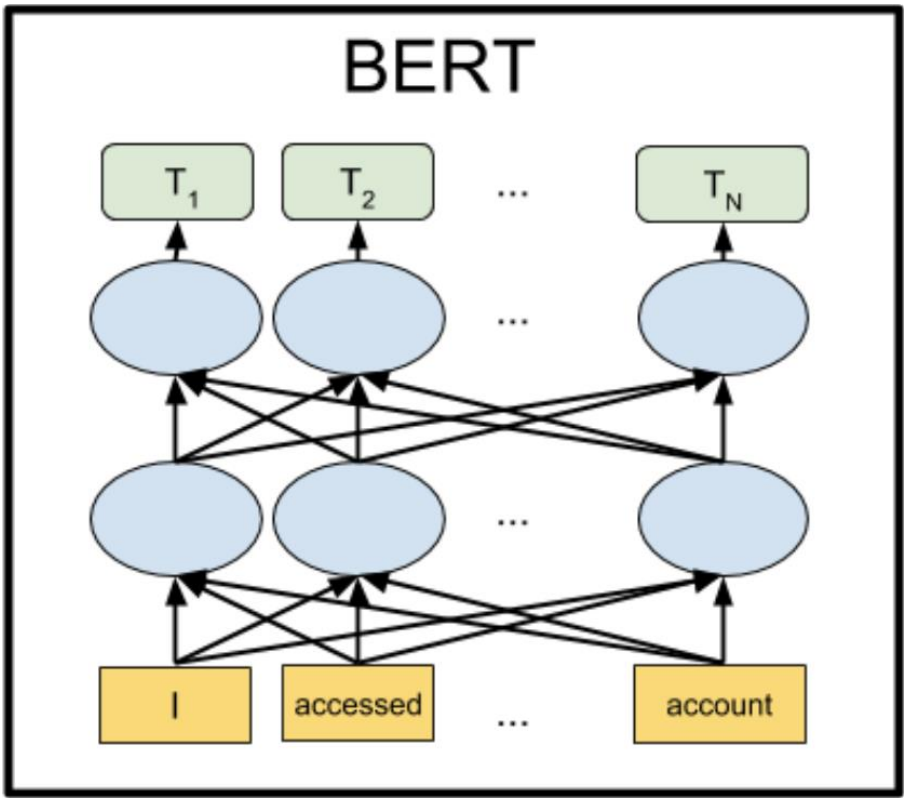

Рисунок 2.11 – BERT-модель [30]

GPT-2 — це мовна модель на основі трансформатора глибокого навчання без контролю, створена OpenAI ще в лютому 2019 року з єдиною метою — передбачити наступне слово (слова) у реченні. GPT-2 — це абревіатура від «Generative Pretrained Transformer 2». Модель має відкритий вихідний код і навчається на більш ніж 1,5 мільярдах параметрів, щоб створити наступну послідовність тексту для певного речення. Завдяки різноманітності набору даних, який використовується в процесі навчання, ми можемо отримати адекватну генерацію тексту для тексту з різних доменів. GPT-2 у 10 разів більше параметрів і в 10 разів більше даних, ніж його попередник GPT [32]

Навчання мовної моделі на великому та різноманітному наборі даних:

– вибір веб-сторінок, які курували/відфільтрували люди;

– очищення та видалення дублікатів текстів, а також видалення всіх документів Вікіпедії, щоб мінімізувати збіги навчальних і тестових наборів;

– використовуючи отриманий набір даних WebText із трохи більше 8 мільйонів документів із загальним обсягом 40 ГБ тексту.

Використання версії кодування пари байтів (BPE) на рівні байтів для представлення вхідних даних [32].

Створення дуже великої моделі на основі трансформатора GPT-2:

– найбільша модель включає 1542M параметрів і 48 шарів;

– модель в основному відповідає моделі OpenAI GPT з невеликими модифікаціями (тобто розширення словникового запасу та розміру контексту, зміна ініціалізації тощо) (рис. 2.12).

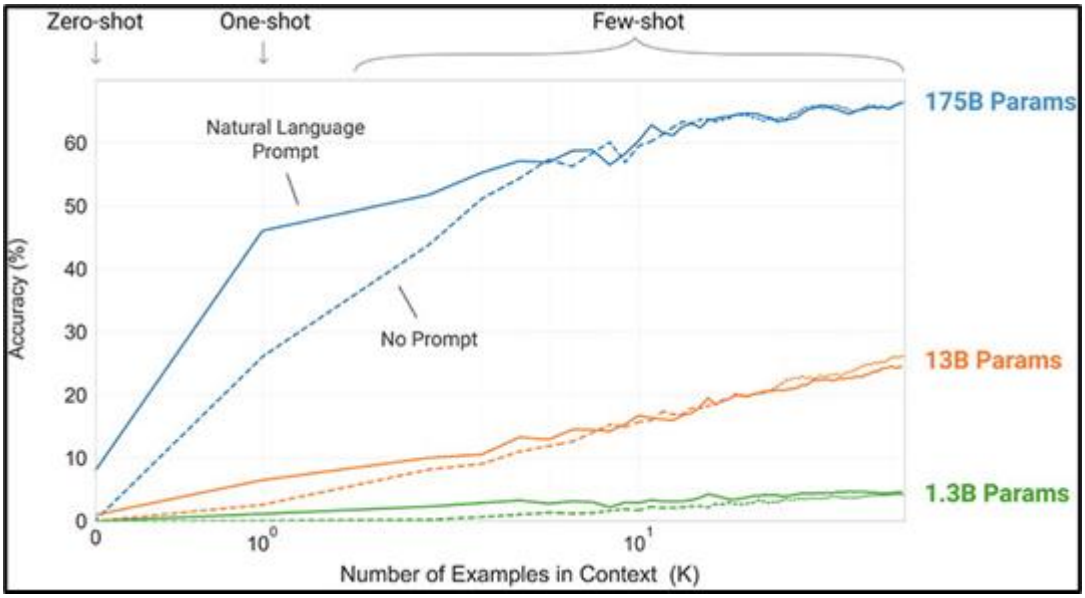

Рисунок 2.12 – Кількість параметрів в GPT-моделях [31]

Мовні завдання, такі як читання, підсумовування та переклад, можуть вивчатися за допомогою GPT-2 із необробленого тексту без використання навчальних даних для конкретної області.

Існують обмеження, які необхідно враховувати при роботі з генеруванням природної мови. Це активна галузь досліджень, але ця галузь

перебуває в зародковому стані, щоб подолати її обмеження. Обмеження включають повторюваний текст, неправильне розуміння вузькотехнічних і спеціалізованих тем і неправильне розуміння контекстних фраз.

Мова та лінгвістика — це складна сфера, яка, як правило, вимагає від людини років навчання та досвіду, щоб зрозуміти не лише значення слів, але й те, як складати речення та давати контекстуально значущі відповіді та використовувати відповідний сленг. Це також можливість створювати індивідуальні та масштабовані моделі для різних доменів. Прикладом, наданим OpenAI, є навчання GPT-2 за допомогою набору даних Amazon Reviews, щоб навчити модель писати відгуки на основі таких речей, як рейтинг і категорія (рис. 2.13) [32].

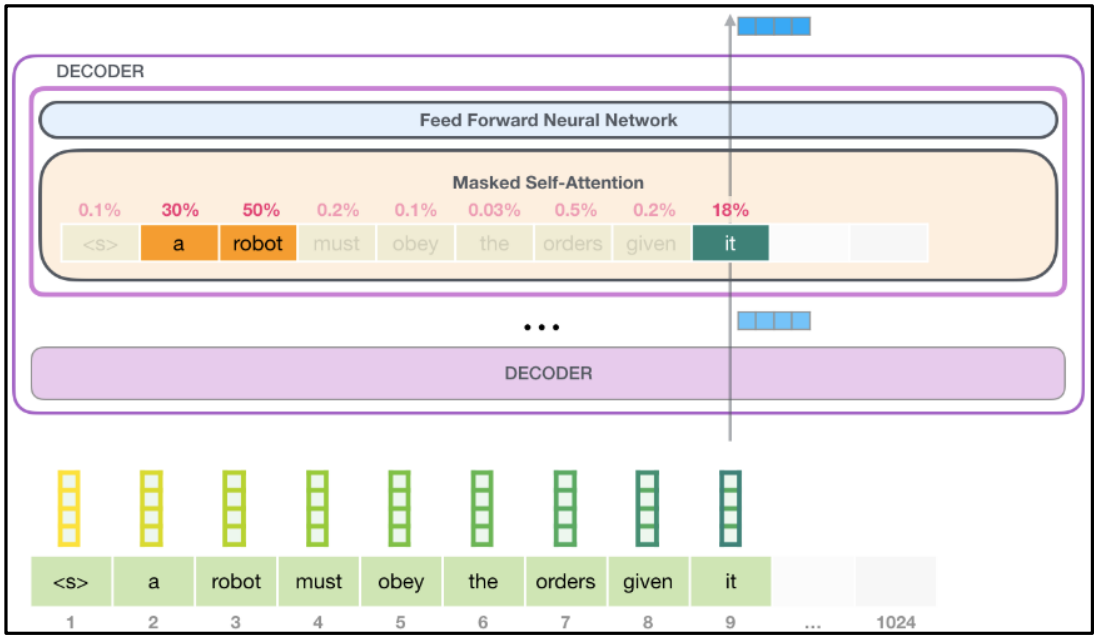

Рисунок 2.13 – Модель GPT2

XLNet. Дослідники з Університету Карнегі-Меллона та Google розробили нову модель XLNet для завдань обробки природної мови (NLP), таких як розуміння прочитаного, класифікація тексту, аналіз настроїв та інші. Завдяки можливості моделювання двонаправлених контекстів попереднє навчання на основі зняття шуму з автокодуванням, як BERT, забезпечує кращу продуктивність, ніж підходи до попереднього навчання, засновані на

авторегресійному моделюванні мови. Однак, покладаючись на спотворення вхідних даних за допомогою масок, BERT нехтує залежністю між замаскованими позиціями та страждає від невідповідності попереднього навчання та точного налаштування. У світлі цих плюсів і мінусів пропонується XLNet, узагальнений авторегресійний метод попереднього навчання, який дозволяє вивчати двонаправлені контексти шляхом максимізації очікуваної ймовірності для всіх перестановок порядку факторизації та долає обмеження BERT завдяки його авторегресії формулювання. Крім того, XLNet інтегрує ідеї з Transformer-XL, найсучаснішої моделі авторегресії, у попередню підготовку. Емпірично XLNet перевершує BERT у 20 завданнях (рис. 2.14) [33].

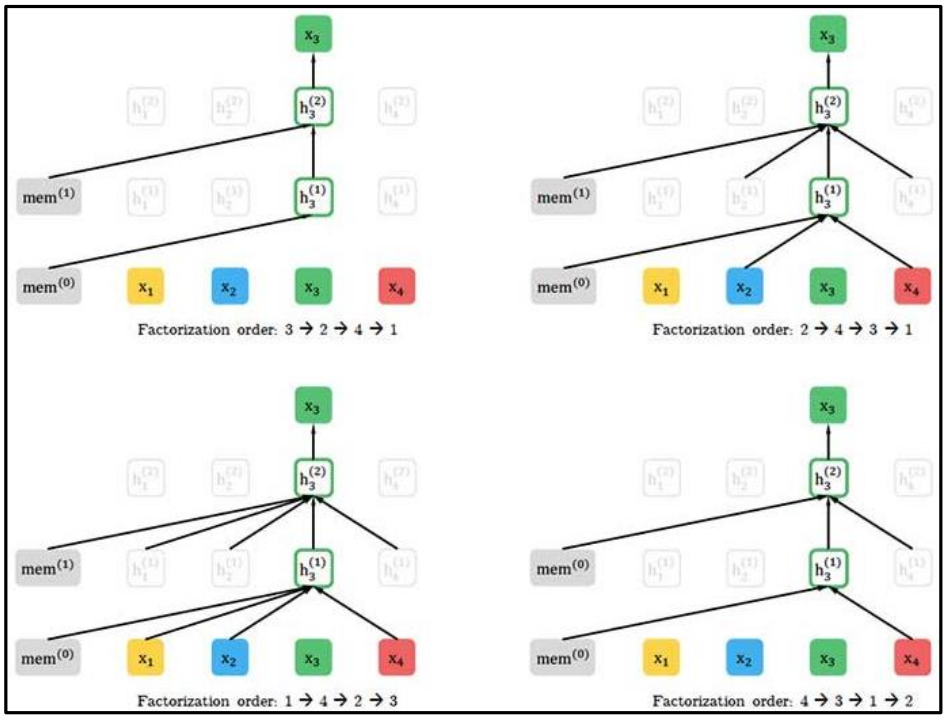

Рисунок 2.14 – XLNet-модель

XLNet використовує операцію перестановки під час навчання, що дозволяє контексту складатися з токенів як зліва, так і справа, захоплюючи двонаправлений контекст, роблячи його узагальненою моделлю мови AR з урахуванням порядку. Під час попереднього навчання XLNet приймає

сегментний рекурентний механізм і схему відносного кодування, запропоновані в Transformer-XL.

По суті, нова ціль моделювання мови перестановок (додаткову інформацію див. у статті) дозволяє спільно використовувати параметри моделі в усіх переставлених порядках факторизації. Це дає змогу моделі AR належним чином і ефективно фіксувати двонаправлений контекст, уникаючи припущення незалежності та розбіжності перед навчанням і тонким налаштуванням, яким зазнає BERT [33].

Простіше кажучи, XLNet зберігає оригінальний порядок послідовності, використовує позиційне кодування та покладається на спеціальну маску уваги в Transformers для досягнення згаданої перестановки порядку факторизації. Іншими словами, оригінальну архітектуру Transformer модифікують і повторно параметризують, щоб уникнути таких проблем, як двозначність цілі та невідповідність попереднього налаштування.

Основні зміни відбуваються в шарах прихованого представлення. XLNet базується на Transformer-XL, який використовується як основна структура попереднього навчання. Очевидно, щоб запропонована операція перестановки працювала, пропонується кілька модифікацій, які забезпечують належне повторне використання прихованих станів із попередніх сегментів. Деякі ідеї дизайну від BERT також використовуються для виконання часткового прогнозування та підтримки певних завдань, які складаються з кількох сегментів, як-от запитання та контекстний абзац у відповіді на запитання.

Завдяки можливості моделювання двонаправлених контекстів попереднє навчання на основі зняття шуму з автокодуванням, як BERT, забезпечує кращу продуктивність, ніж підходи до попереднього навчання, засновані на авторегресійному моделюванні мови. Однак, покладаючись на спотворення вхідних даних за допомогою масок, BERT нехтує залежністю між замаскованими позиціями та страждає від невідповідності попереднього навчання та точного налаштування. У світлі цих плюсів і мінусів ми

пропонуємо XLNet, узагальнений авторегресійний метод попереднього навчання, який дозволяє вивчати двонаправлені контексти шляхом максимізації очікуваної ймовірності для всіх перестановок порядку факторизації та долає обмеження BERT завдяки його авторегресії формулювання. Крім того, XLNet інтегрує ідеї з Transformer-XL, найсучаснішої моделі авторегресії, у попередню підготовку. Емпірично XLNet перевершує BERT у 20 завданнях, часто з великим відривом, і досягає найсучасніших результатів у 18 завданнях, включаючи відповіді на запитання, висновок природної мови, аналіз настроїв і ранжування документів (рис. 2.15) [33].

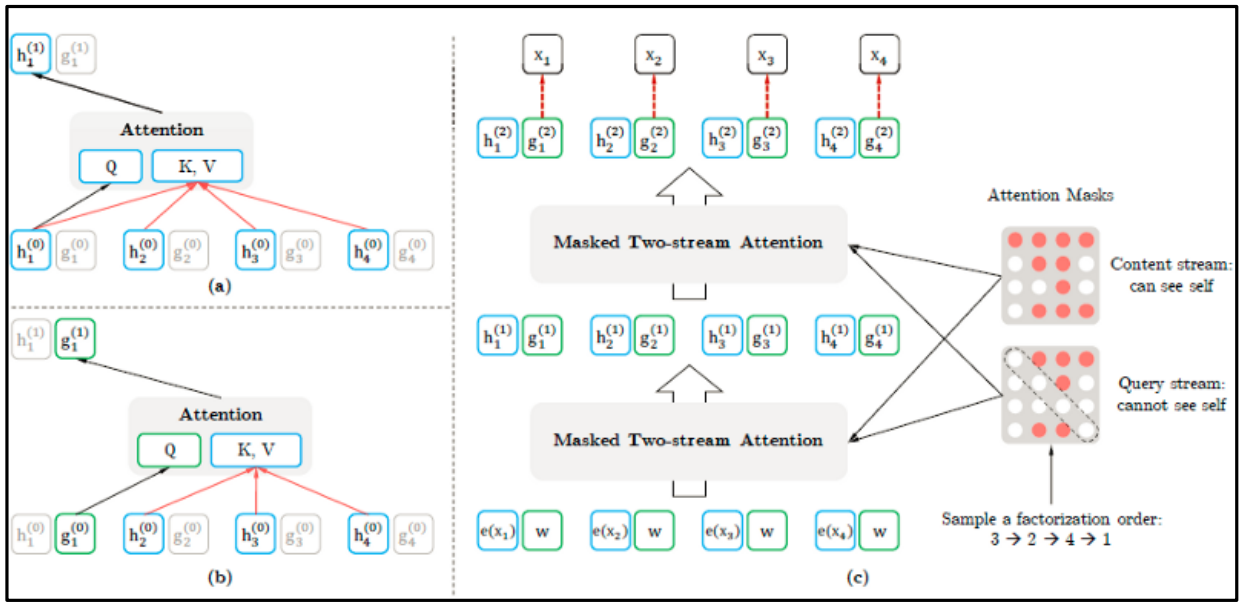

Рисунок 2.15 – XLNet-модель

GPT-3 — це «генеративний попередньо навчений трансформатор», який є 3-ю версією та оновленою версією GPT-2. Версія 3 виводить модель GPT на абсолютно новий рівень, оскільки вона навчена колосальним 175 мільярдам параметрів (що в 10 разів перевищує розмір її попередника GPT-2). GPT-3 був навчений на наборі даних з відкритим кодом під назвою « Common Crawl » та інших текстах з OpenAI, таких як записи у Вікіпедії [32].

GPT-3 був створений, щоб бути надійнішим, ніж GPT-2, оскільки він здатний обробляти більше нішевих тем. Відомо, що GPT-2 має низьку продуктивність, коли йому давали завдання в таких спеціалізованих областях, як музика та оповідання. GPT-3 тепер може піти далі з такими завданнями, як відповіді на запитання, написання есе, узагальнення тексту, мовний переклад і генерація комп'ютерного коду.

Довгий час багато програмістів хвилювалися через те, що їх замінять штучним інтелектом, і тепер, схоже, це стане реальністю. У міру того, як глибокі фейкові відео набирають популярності, мова й текст також керуються ШІ, щоб імітувати людей. Невдовзі під час розмови по телефону чи спілкування в Інтернеті (наприклад, у програмах для чату) може бути важко визначити, чи розмовляєте ви з реальною людиною чи зі штучним інтелектом. [32].

GPT-3 має таку саму архітектуру на основі уваги, що й GPT-2 (рис. 2.16).

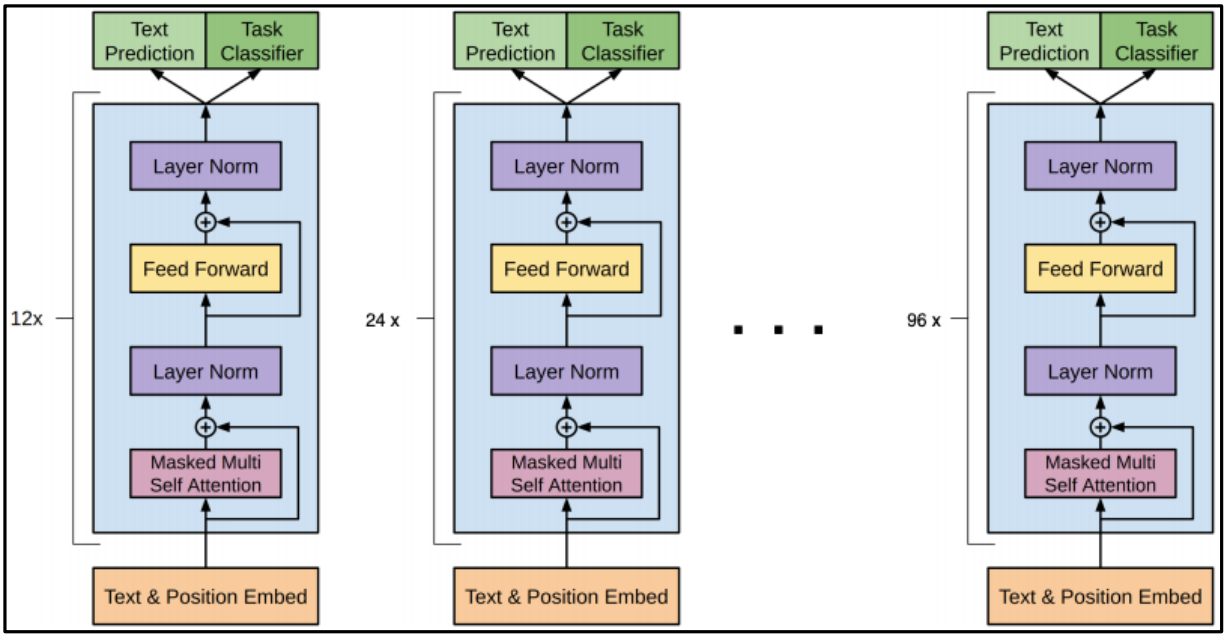

Рисунок 2.16 – Модель GPT-3 [32]

PaLM. Дослідницька команда Google зробила великий внесок у розробку попередньо підготовлених мовних моделей із своїми моделями BERT, ALBERT і T5. Одним із їхніх останніх внесків є Pathways Language Model

(PaLM), модель трансформатора з 540 мільярдами параметрів, що містить лише щільний декодер, навчена за допомогою системи Pathways. Метою системи Pathways є оркестрування розподілених обчислень для прискорювачів. З його допомогою команда змогла ефективно навчити одну модель на кількох модулях TPU v4. Експерименти з сотнями завдань із розуміння мови та її генерування продемонстрували, що PaLM досягає найсучаснішої швидкодії за декілька кроків у більшості завдань із проривними можливостями, продемонстрованими в розумінні мови, генерації мови, міркуванні та завданнях, пов'язаних із кодом. PaLM працює нарівні з точно налаштованим Codex 12B, використовуючи в 50 разів менше коду Python для навчання, підтверджуючи, що великі мовні моделі ефективніше передають навчання з інших мов програмування та даних природної мови (рис. 2.17) [33].

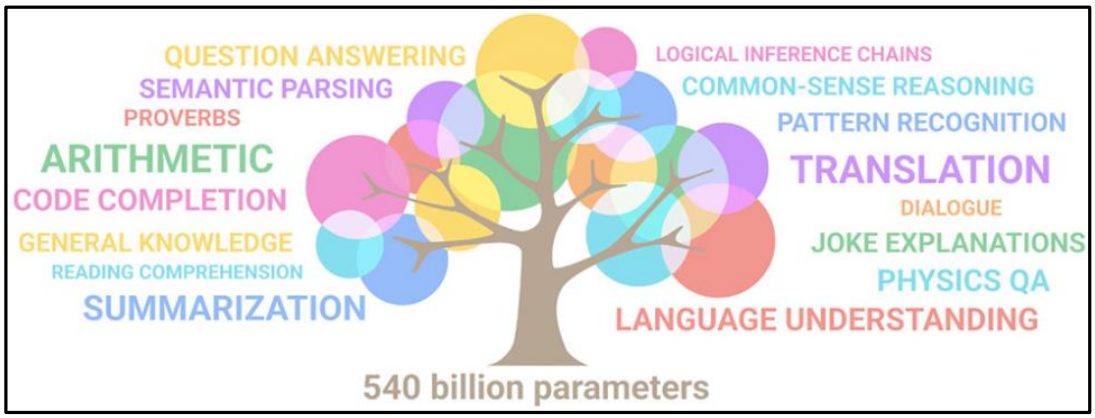

Рисунок 2.17 – Модель PaLM

T5 (Text-to-Text Transfer Transformer) — слугує головним чином як код для відтворення експериментів у дослідженні меж трансферного навчання за допомогою уніфікованого перетворювача тексту в текст. Дослідницька група Google пропонує єдиний підхід до передачі навчання в NLP з метою встановлення нового рівня мистецтва в цій галузі. З цією метою вони пропонують розглядати кожну проблему NLP як проблему «перетворення тексту в текст». Така структура дозволяє використовувати ту саму модель, ціль, процедуру навчання та процес декодування для різних завдань,

включаючи підсумовування, аналіз настроїв, відповіді на запитання та машинний переклад. Дослідники називають свою модель перетворювачем передачі тексту в текст (T5) і тренують її на великому масиві даних, зібраних з Інтернету, щоб отримати найсучасніші результати для ряду завдань NLP [31].

Переглянемо структуру моделі T5 (рис. 2.18).

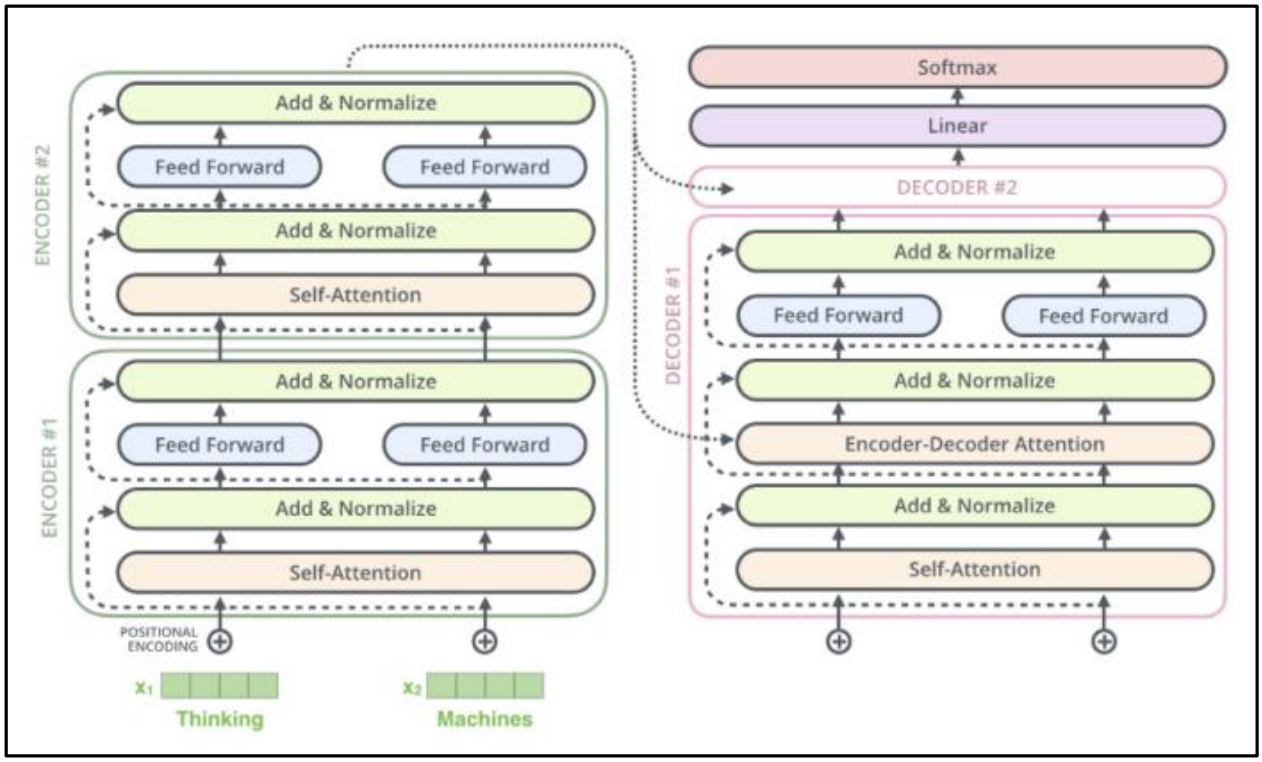

Рисунок 2.18 – Структура моделі T5 [31]

Структура режиму є просто стандартним трансформатором кодера-декодера.

Документ містить кілька важливих внесків:

– забезпечення всебічної точки зору на те, де знаходиться сфера NLP, досліджуючи та порівнюючи існуючі техніки;

– представляємо новий підхід до перенесення навчання в NLP, пропонуючи розглядати кожну проблему NLP як завдання з тексту в текст;

– модель розуміє, які завдання слід виконати, завдяки префіксу завдання, доданому до оригінального вхідного речення (наприклад, «перекласти з англійської на німецьку:», «резюмувати:»);

– представляємо та випускаємо новий набір даних, що складається з сотень гігабайт чистого англійського тексту, зібраного в Інтернеті, Колосальний чистий сканований корпус (C4);

– навчання великої (до 11B параметрів) моделі під назвою «Трансформатор передачі тексту в текст» (T5) на наборі даних C4.

Напрямки майбутніх досліджень:

– дослідження методів досягнення кращої продуктивності з дешевшими моделями;

– вивчення більш ефективних методів вилучення знань;

– подальше дослідження мовно-агностичних моделей.

Переглянемо модель T5 (рис. 2.19).

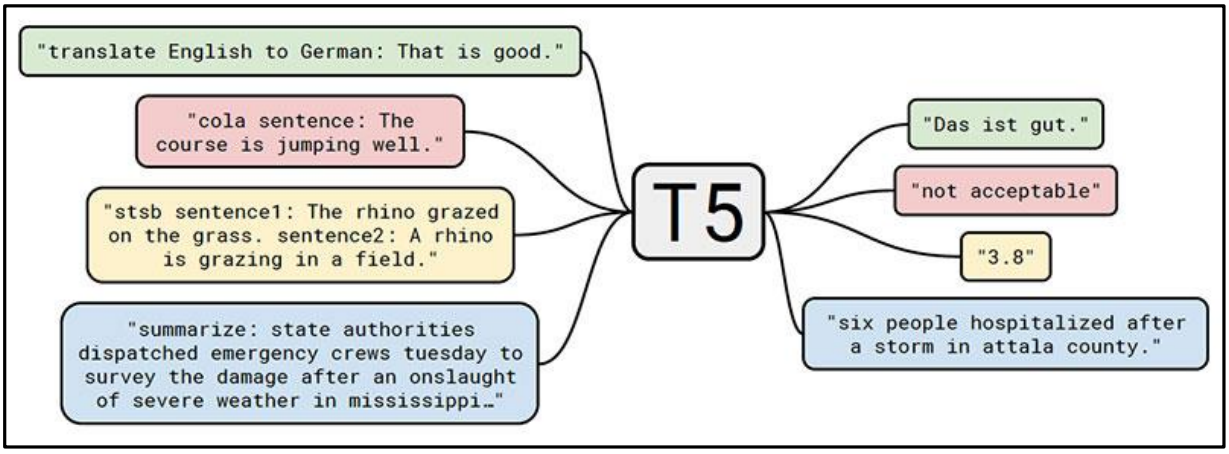

Рисунок 2.19 – Модель T5 [31]

Також потрібно обрати бібліотеки для роботи з анотуванням:

«Bert Extractive Summarizer» — це трансформатор, який використовується для подолання обмежень RNN та інших нейронних мереж як довгострокових залежностей. Це попередньо навчена модель, яка природно двонаправлена. Цю попередньо навчену модель можна налаштувати для легкого виконання завдань NLP, як зазначено, у випадку підведення підсумків [34].

На даний момент BERT вважається найкращою доступною технікою для виконання завдань NLP.

Головними перевагами є:

– моделі BERT попередньо навчені на величезних наборах даних, тому подальше навчання не потрібне;

– він використовує потужну плоску архітектуру з шарами перетворення речень, щоб отримати найкращі результати під час узагальнення.

Вважається, що підсумкові речення представляють найважливіші моменти документа.

Для набору речень {send1,sent2,sent3,...,sentn,} є дві можливості, тобто yi={0,1}, що вказує, чи буде вибрано конкретне речення чи ні (рис. 2.20)

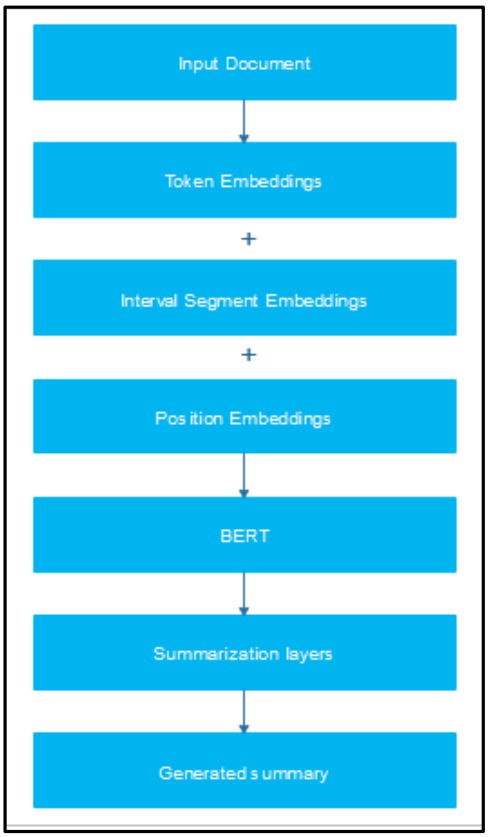

Рисунок 2.20 – Методологія BERT-моделі [34]

Будучи навченим як маскована модель, вихідні вектори позначаються маркерами замість речень. На відміну від інших екстрактивних підсумків, він використовує вбудовування для позначення різних речень і має лише дві мітки, а саме речення A та речення B, а не кілька речень. Ці вбудовування відповідно змінено для створення необхідних підсумків (рис. 2.21).

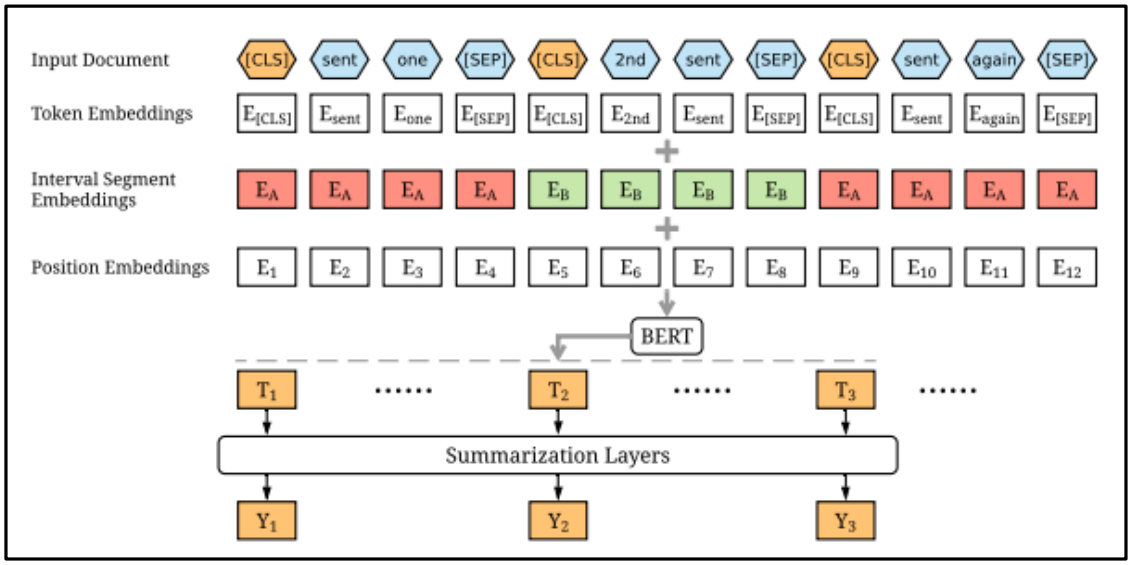

Рисунок 2.22 - Огляд архітектури моделі Bertsum [34]

Весь процес можна розділити на кілька етапів, а саме [34]:

– кодування кількох речень — на цьому кроці речення з вхідного документа кодуються для попередньої обробки. Кожному реченню передує тег CLS, а після нього – тег SEP. Тег CLS використовується для агрегування характеристик одного або кількох речень;

– вбудовування інтервальних сегментів — цей крок призначений для розрізнення речень у документі. Реченням присвоюється одна з міток, розглянутих вище.

«Transformers» — це модель глибокого навчання, представлена в 2017 році, яка використовується в основному в області обробки природної мови (NLP). Як і рекурентні нейронні мережі (RNN), Transformers розроблено для обробки послідовних даних, таких як природна мова, для таких завдань, як переклад і резюмування тексту. Однак, на відміну від RNN, Transformers не

вимагає, щоб послідовні дані оброблялися по порядку. Наприклад, якщо вхідними даними є речення природною мовою, Transformers не потрібно обробляти його початок перед кінцем. Завдяки цій функції Transformer дозволяє набагато більше розпаралелювати, ніж RNN, і тому скорочує час навчання та широко використовується для моделювання мови [35].

Модель Transformer складається з кодера та декодера – кожен є стеком того, що ми можемо назвати блоками трансформатора. Ця архітектура була доцільною, оскільки модель вирішувала машинний переклад – проблему, коли архітектури кодера-декодера були успішними в минулому (рис. 2.23)

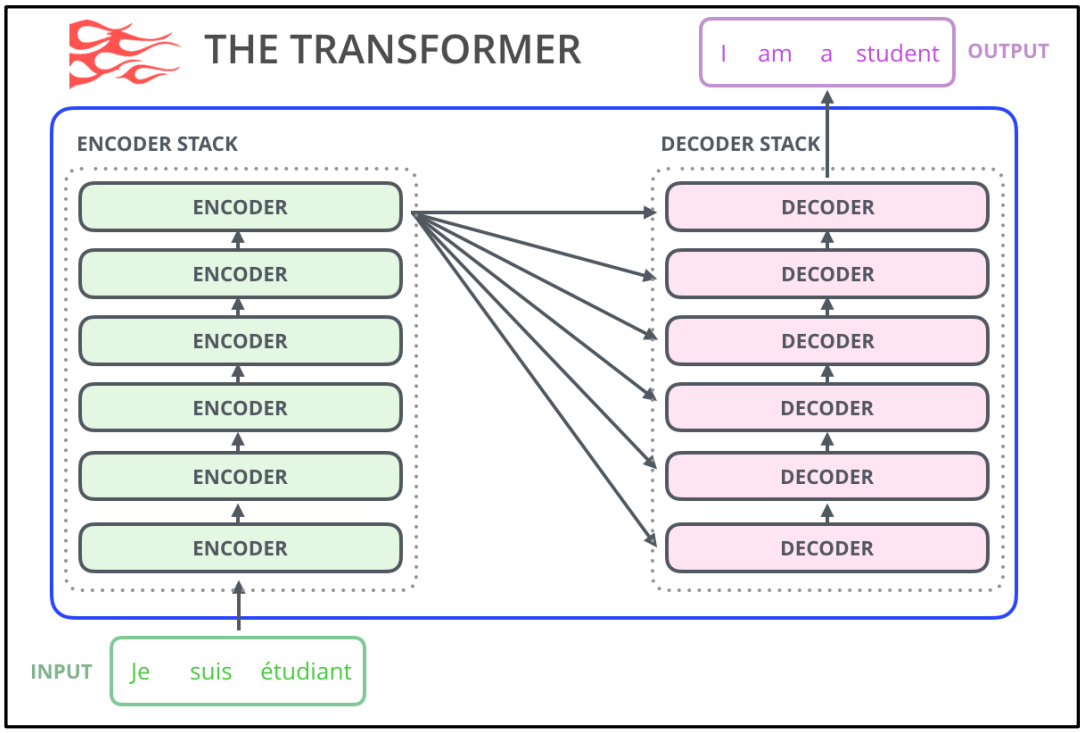

Рисунок 2.23 – Модель Transformers [35]

«spaCy» — безкоштовна бібліотека з відкритим кодом для обробки природної мови на Python. Серед безлічі бібліотек NLP сьогодні spaCy справді виділяється сама по собі. Ця бібліотека є багатофункціональною та гнучкою. Факторами, які працюють на користь spaCy, є набір функцій, які він пропонує, простота використання та той факт, що бібліотека завжди підтримується в актуальному стані [36].

Це одна з двох найпопулярніших бібліотек для NLP, іншою є NLTK. Багаті функції, які пропонує spaCy, роблять його чудовим вибором для NLP, вилучення інформації та розуміння природної мови. Ключовою перевагою spaCy є те, що він розроблений для оптимальної та надійної роботи з великими обсягами даних.

Першим кроком для текстового рядка під час роботи зі spaCy є передача його в об'єкт NLP. Цей об'єкт, по суті, є конвеєром кількох операцій попередньої обробки тексту, через які має пройти вхідний текстовий рядок.

spaCy представляє концепцію трубопроводів. Коли пропускаєте текст через конвеєр, він проходить різні етапи (або канали) обробки. Вихід з одного кроку (або труби) подається в наступний крок (або трубу).

spaCy пропонує багато навчених конвеєрів для різних мов. Як правило, навчений конвеєр включає теггер, лемматизатор, синтаксичний аналізатор і розпізнавач сутностей (рис. 2.24).

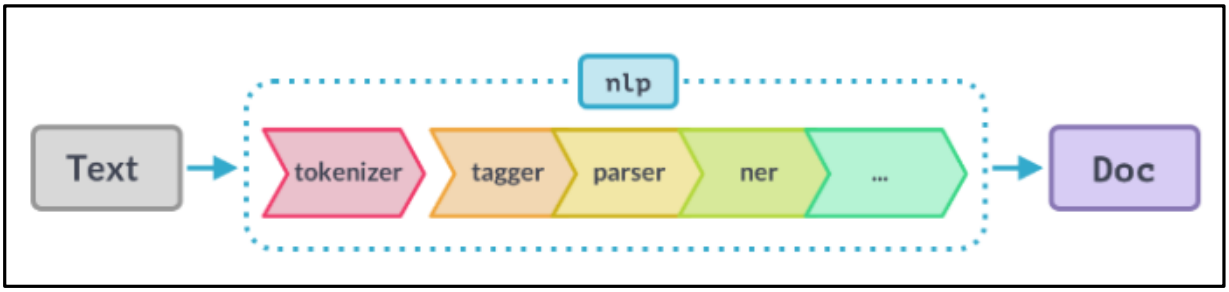

Рисунок 2.24 – Конвеєр NLP [36]

Як можна побачити на малюнку вище, конвеєр NLP має кілька компонентів, таких як токенайзер, теггер, парсер, ner тощо. Отже, вхідний текстовий рядок має пройти через усі ці компоненти, перш ніж ми зможемо над ним працювати.

NLTK є провідною платформою для створення програм Python для роботи з даними людської мови. Він надає прості у використанні інтерфейси для більш ніж 50 корпусів і лексичних ресурсів, таких як WordNet, а також набір бібліотек для обробки тексту для класифікації, токенізації, сформування

основи, тегування, синтаксичного аналізу та семантичного міркування, оболонки для індустріальних бібліотек NLP, і активний дискусійний форум [37].

NLTK називають «чудовим інструментом для навчання та роботи в комп'ютерній лінгвістиці з використанням Python» і «дивовижною бібліотекою для гри з природною мовою».

Обробка природної мови за допомогою Python забезпечує практичний вступ до програмування для обробки мови. Написана творцями NLTK, вона веде читача через основи написання програм на Python, роботи з корпусами, категоризації тексту, аналізу лінгвістичної структури тощо.

NLTK означає Natural Language Toolkit. Це набір бібліотек і програм для символічного та статистичного NLP для англійської мови. Він поставляється з графічними демонстраціями та зразками даних [38].

Вперше побачивши світ у 2001 році, NLTK сподівається підтримувати дослідження та навчання в NLP та інших сферах, тісно пов'язаних із цим. До них відносяться штучний інтелект, емпірична лінгвістика, когнітивна наука, пошук інформації та машинне навчання.

Інтеграція лексичного корпусу (WordNet, Stopwords тощо), токенізація, аналіз настроїв (рис. 2.25).

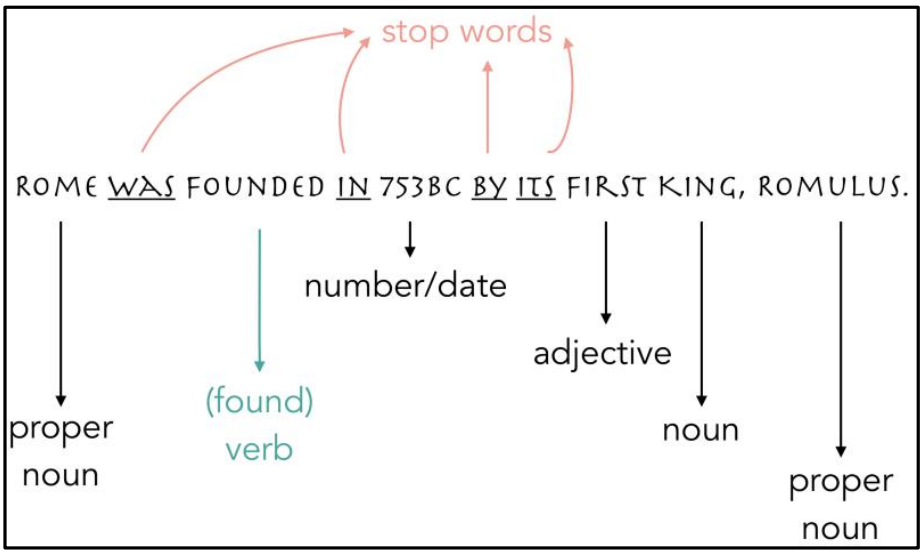

Рисунок 2.25 – Інтеграція лексичного корпусу NLTK [38]

Natural Language Toolkit (NLTK) — це найбільша бібліотека обробки природної мови, яка підтримує багато мов. Порівняємо «NLTK» і «spaCy» у таблиці 2.1 [38].

Таблиця 2.1 – Порівняння бібліотек «NLTK» і «spaCy»

Номер	NLTK	spaCy
1	NLTK в основному призначений для досліджень.	spaCy призначений для виробничого використання.
2	NLTK підтримує багато мов.	Наразі spaCy надає навчені конвеєри для 23 мов і підтримує понад 66 мов.
3	NLTK використовує підхід до обробки рядків і має модульну архітектуру.	spaCy дотримується об'єктно-орієнтованого підходу.
4	NLTK надає велику кількість різних алгоритмів NLP і, отже, є кращим для дослідження та створення інноваційних рішень. Користувач може вибрати певний алгоритм з доступних варіантів для конкретного завдання.	spaCy використовує найкращий алгоритм для конкретного завдання. Користувачеві не потрібно вибирати алгоритм.
5	NLTK може бути повільнішим.	spaCy оптимізовано для швидкості.
6	Вона створена за допомогою Python.	Вона створена за допомогою Python.

Gensim — найшвидша бібліотека для навчання векторних вставок – Python. Основні алгоритми в Gensim використовують загартовані в боях,

високо оптимізовані та розпаралелені процедури C. Gensim може обробляти доволі великі корпуси, використовуючи алгоритми потокової передачі даних. Немає обмежень "набір даних повинен поміститися в оперативну пам'ять". Працює на Linux, Windows і OS X, а також на будь-якій іншій платформі, яка підтримує Python і NumPy. Gensim є однією з найдосконаліших бібліотек ML. Спільнота Gensim також публікує попередньо підготовлені моделі для певних сфер, як-от право чи охорона здоров’я, через проект Gensim-data [39].

Gensim — це бібліотека Python з відкритим вихідним кодом, яку можна використовувати для моделювання тем, індексування документів, а також для схожості з великими корпусами. Алгоритми Gensim не залежать від пам’яті щодо розміру корпусу. Він також був розроблений для розширення з іншими алгоритмами векторного простору.

Gensim забезпечує реалізацію алгоритму Word2Vec разом з деякими іншими функціями обробки природної мови в класі Word2Vec. Давайте подивимося, як створити модель Word2Vec за допомогою Gensim.

Word2vec — це інструмент, який створює вбудовування слів: на основі вхідного тексту він створить векторне представлення кожного слова. Word2vec спочатку був реалізований у Google Томашем Міколовим, але сьогодні можна знайти багато інших реалізацій.

Щоб створити вбудовування слів, word2vec від Gensim використовує нейронну мережу з одним прихованим шаром. Введенням є кожне слово разом із налаштованим контекстом (зазвичай від 5 до 10 слів). Ви повинні навчити цю нейронну мережу передбачати слово за його контекстом або навпаки (передбачати контекст за словом). Це залежить від режиму навчання, але давайте проілюструємо перший підхід: скажімо, у вас є слово «solr» зі «introduction to solr», і ми намагаємося передбачити його за допомогою трьох нейронів (рис 2.26) [40].

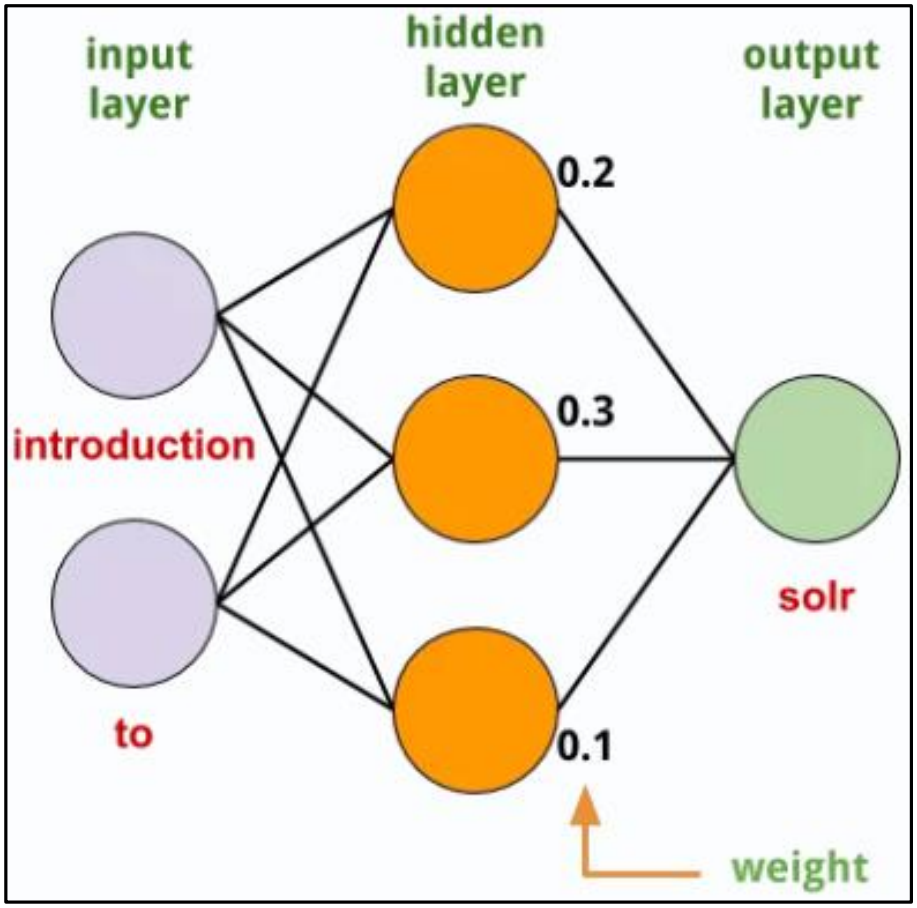

Рисунок 2.26 – Передбачення Word2Vec with Gensim

За результатом аналізу різних моделей було обрано три NLP-моделі: BERT, GPT-2 та XLNet. Тому що з представлених моделей у моделі GPT-3 є вражаючі здібності генерувати текст, але йому бракує адекватності, при роботі можна з’ясувати, що, попри грамотність та вірну стилістику речень, контекст часто спотворений.

Для розгортання яких доцільно використовувати бібліотеки «Bert Extractive Summarizer», «Transformers», «spaCy» для проведення адекватного анотування.

2.4 Висновки

У даному розділі проведено огляд датасету, здійснено розвідувальний аналіз по параметру «Забруднення», а саме – побудовано різні графіки по

різним критеріям, проведено вибір оптимальних інформаційних технологій для роботи.

Проведено аналіз сучасних мовних моделей і технологій штучного інтелекту і як оптимальні вибрано NLP-моделі: BERT, GPT-2, XLNet. Для розгортання яких доцільно використовувати бібліотеки «Bert Extractive Summarizer», «Transformers», «spaCy» для проведення адекватного анотування.

3 СТВОРЕННЯ ІНТЕЛЕКТУАЛЬНОЇ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ АВТОМАТИЧНОГО СИНТЕЗУ АНОТАЦІЙ ДО МАТЕРІАЛІВ ПРО ВОДНІ РЕСУРСИ МЕТОДАМИ ШТУЧНОГО ІНТЕЛЕКТУ

3.1 Розроблення інформаційної технології

Для початку побудуємо алгоритм роботи інтелектуальної технології (рис. 3.1).

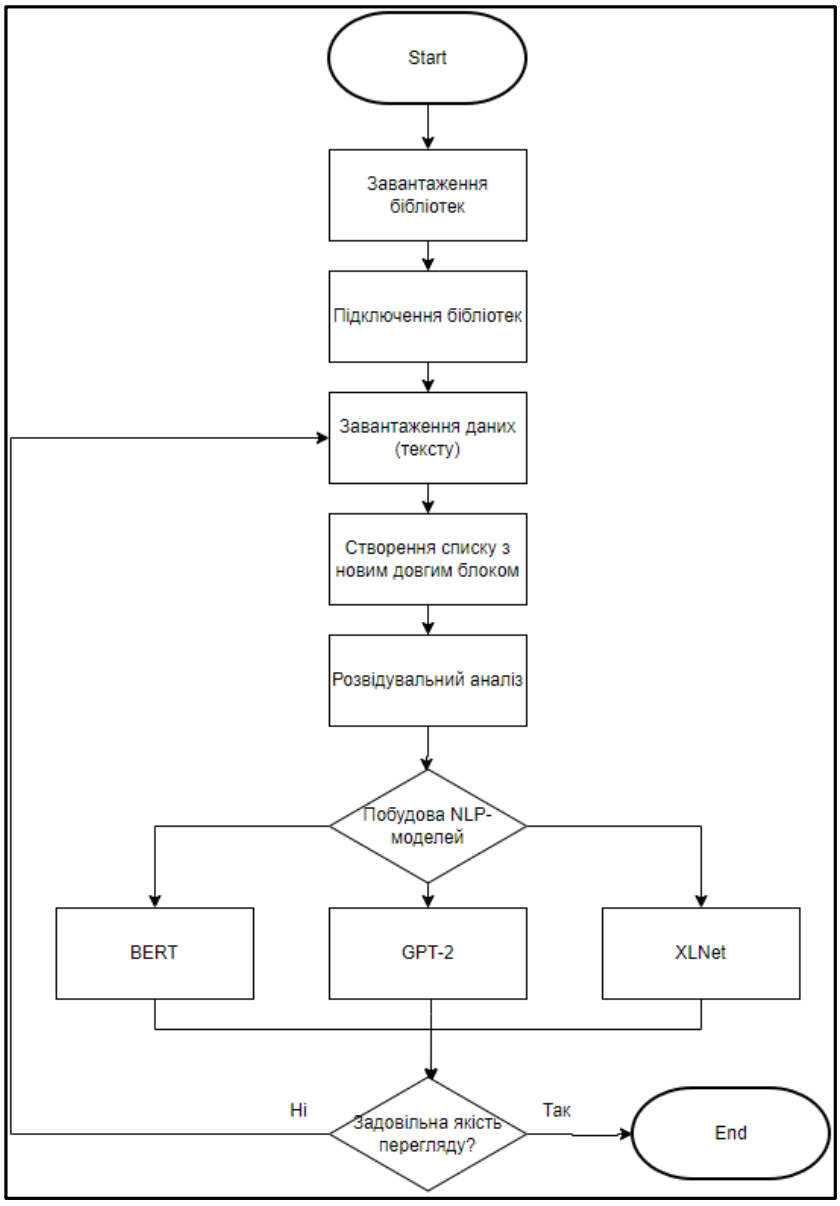

Рисунок 3.1 – Алгоритм роботи програми

Після побудови алгоритму розпочнемо його виконання. Інсталюємо необхідні для роботи компоненти/бібліотеки які було обрано розділом вище (рис. 3.2).

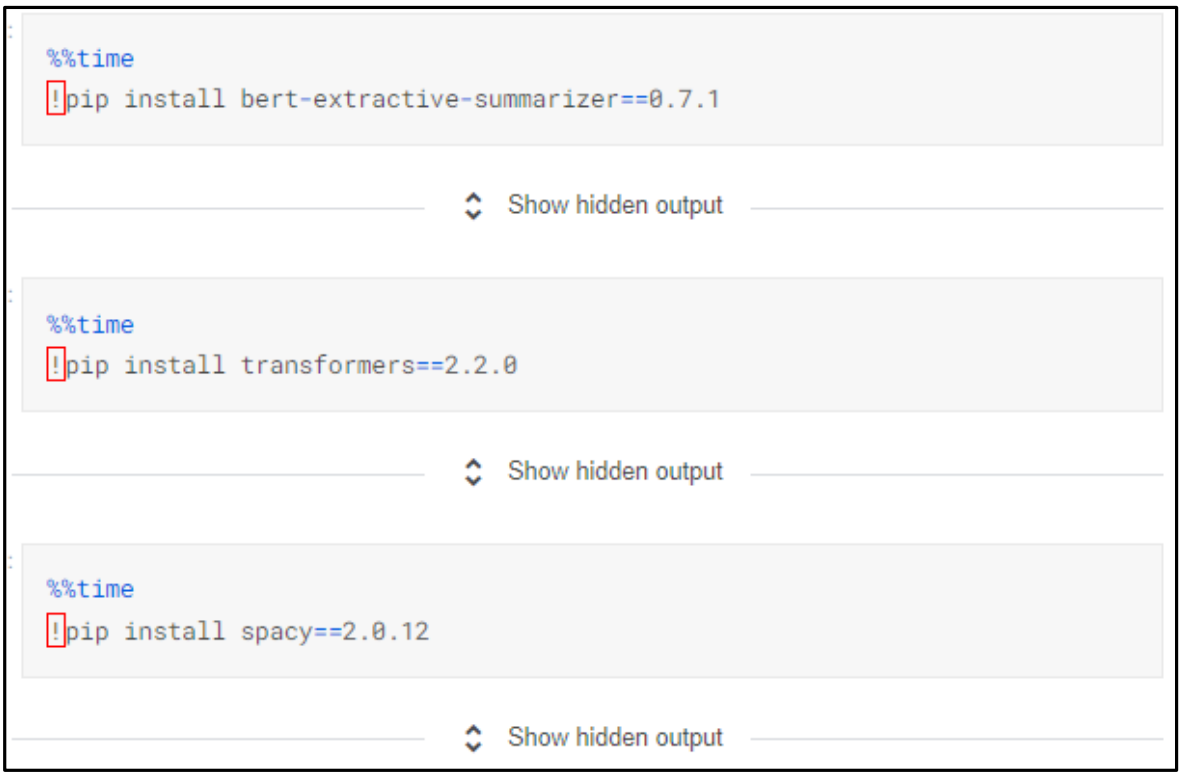

Рисунок 3.2 – Інсталяція компонентів

Також імпортуємо необхідні для роботи бібліотеки, ключовою з яких є «summarizer» для скорочення тексту (анотування) (рис. 3.3).

```
import numpy as np
import pandas as pd
import seaborn as sns

import matplotlib
import matplotlib.pyplot as plt

from summarizer import Summarizer,TransformerSummarizer
from sklearn.metrics import accuracy_score

import warnings
warnings.filterwarnings('ignore')
```

Рисунок 3.3 – Імпорт бібліотек

Наступним кроком завантажимо текст, який будемо аналізувати та проводити сумарайзинг (анотування). Текст беремо з англомовного csv файлу «water_problem_nlp_en_for_Kaggle_100.csv» (рис. 3.4).

```
# Thanks to https://www.kaggle.com/vbmokin/nlp-for-en-bert-classification-for-water-report
df = pd.read_csv('../input/nlp-reports-news-classification/water_problem_nlp_en_for_Kaggle_100.csv', delimiter=';', header=0)
df = df.fillna(0)

convert_dict = {'text': str,
                'env_problems': int,
                'pollution': int,
                'treatment': int,
                'climate': int,
                'biomonitoring': int}

df = df.astype(convert_dict)
df = df[:5]
df
```

	text	env_problems	pollution	treatment	climate	biomonitoring
0	In the Southern Bug basin wastewater is discha...	1	1	0	0	0
1	This amount is distributed as follows: urban a...	1	1	0	0	0
2	Despite the similar volumes of discharged wast...	1	1	0	0	0
3	They bring 84% of organic pollution 86% of pho...	1	1	0	0	0
4	The input of the industry is between 7–21% and...	1	1	0	0	0

Рисунок 3.4 – Завантаження тексту

Далі створимо список з новим довгим блоком, тому що наш текст був поділений на рядки в csv файлі (рис. 3.5).

```
# Creation the list with new long block
max_length = 400  # minimum characters in each block
i = 0
bodies = []
while i < len(df):
    body = ""
    body_empty = True
    while (len(body) < max_length) and (i < len(df)):
        if body_empty:
            body = df.loc[i,'text']
            body_empty = False
        else: body += " " + df.loc[i,'text']
        i += 1
    bodies.append(body)
    print("Length of blocks =", len(body))
print(f"\nNumber of text blocks = {len(bodies)}\n")
print("Text blocks:\n", bodies)
```

```
Length of blocks = 410
Length of blocks = 553

Number of text blocks = 2

Text blocks:
 ['In the Southern Bug basin wastewater is discharged from 341 organized sources (Map 5). In 2011 (reference year) 211 mln. m3 was discharged in total. This amount is distributed as follows: urban and
rural housing wastewater treatment facilities (WWTPs - vodocanals) - 70 mln. m3 (57 users) industry - 50 mln. m3 (79 users) agriculture - 90 mln. m3 (193 users) other users - 1 mln. m3 (12 users) (Pict
ure. 2.1). ', 'Despite the similar volumes of discharged wastewater major part of pollutants comes with communal WWTPs. They bring 84% of organic pollution 86% of phosphate ions and 84% of mineral nitr
ogen 91% of ammonia nitrogen 87% nitrate nitrogen and 79% nitrite nitrogen. The input of the industry is between 7–21% and agriculture has the lowest impact on water bodies - 0–6%. Of the 92 urban area
s only 51 localities (55%) have centralized collection of communal waste waters and their monitoring. Among the 2878 villages 6 of them (0.2%) have such a monitoring.']
```

Рисунок 3.5 – Створення блоків

Вкажемо мінімальну довжину тексту з анотацією в символах, щоб отримати певну кількість інформації, а не весь текст чи велике і не зрозуміле скорочення (рис. 3.6).

```
min_length_text = 40
```

Рисунок 3.6 – Мінімальна довжина

Далі напишемо код для сумарайзингу (анотування) BERT-моделі, GPT-2 та XLNet використовуючи бібліотеки «Summarizer», «Bert Extractive Summarizer», «Transformers», «spaCy» (рис. 3.7 – 3.9).

```
%%time
bert_summary = []
for i in range(len(bodies)):
    bert_model = Summarizer()
    bert_summary.append(''.join(bert_model(bodies[i], min_length=min_length_text)))
```

Рисунок 3.7 – Побудова BERT-моделі

```
%%time
gpt_summary = []
for i in range(len(bodies)):
    GPT2_model = TransformerSummarizer(transformer_type="GPT2",transformer_model_key="gpt2-medium")
    gpt_summary.append(''.join(GPT2_model(bodies[i], min_length=min_length_text)))
```

Рисунок 3.8 – Побудова GPT-2-моделі

```
%%time
xlnet_summary = []
for i in range(len(bodies)):
    model = TransformerSummarizer(transformer_type="XLNet",transformer_model_key="xlnet-base-cased")
    xlnet_summary.append(''.join(model(bodies[i], min_length=min_length_text)))
```

Рисунок 3.9 – Побудова XLNet-моделі

3.2 Застосування інформаційної технології

Застосовуємо моделі з розділу 2 до даних з розділу 1 та переглядаємо результати (рис. 3.10 – 3.12). Можна зробити висновки, що адекватність всіх моделей є однаковою, що є доволі цікавим та дивним результатом, тому спробуємо провести синтез іншого тексту для точності дослідження.

```
%%time
print("All Summarizing Results:\n")
for i in range(len(bodies)):
    print("ORIGINAL TEXT:")
    print(bodies[i])
    print("\nBERT Summarizing Result:")
    print(bert_summary[i])
    print("\nGPT-2 Summarizing Result:")
    print(gpt_summary[i])
    print("\nXLNet Summarizing Result:")
    print(xlnet_summary[i])
    print("\n\n")
```

Рисунок 3.10 – Код виведення фінальних результатів

```
ORIGINAL TEXT:
In the Southern Bug basin wastewater is discharged from 341 organized sources (Map
5). In 2011 (reference year) 211 mln. m3 was discharged in total. This amount is di
stributed as follows: urban and rural housing wastewater treatment facilities (WWTP
s - vodocanals) - 70 mln. m3 (57 users) industry - 50 mln. m3 (79 users) agricultur
e - 90 mln. m3 (193 users) other users - 1 mln. m3 (12 users) (Picture. 2.1).

BERT Summarizing Result:
In the Southern Bug basin wastewater is discharged from 341 organized sources (Map
5).

GPT-2 Summarizing Result:
In the Southern Bug basin wastewater is discharged from 341 organized sources (Map
5).

XLNet Summarizing Result:
In the Southern Bug basin wastewater is discharged from 341 organized sources (Map
5).
```

Рисунок 3.11 – Фінальні результати

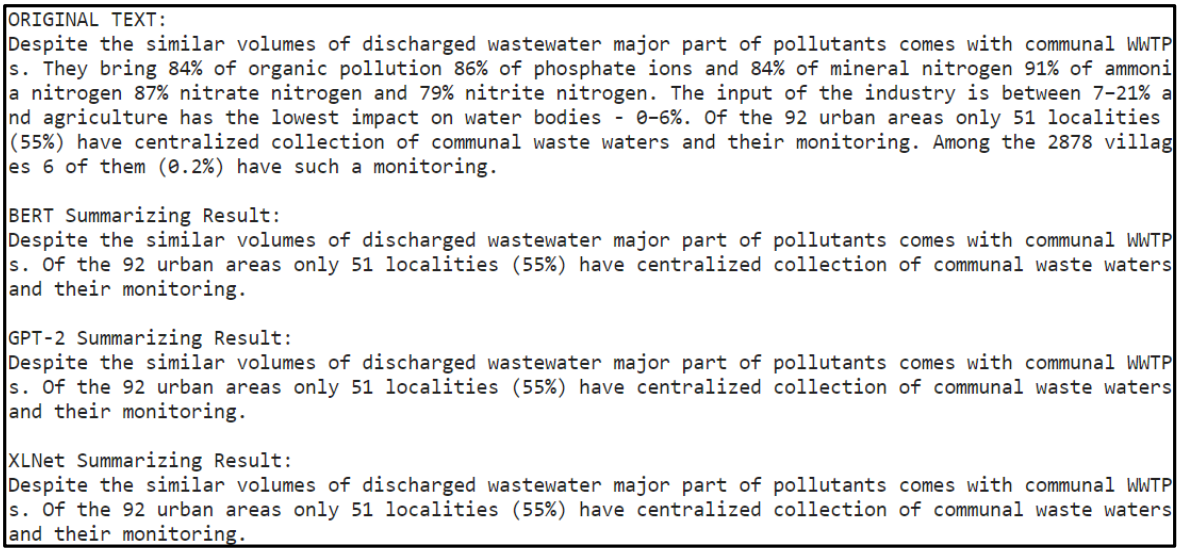

Рисунок 3.12 – Фінальні результати

При синтезі іншого тексту BERT-модель обрала основною частину про результати перевірки екологічного стану, а моделі GPT-2 та XLNet обрали речення з назвою міста, що також могло б бути ключовим в певних ситуаціях.

Тобто, як можна побачити, з даним текстом більш адекватною була BERT-модель, тому що нею було обрано більш узагальнюючий текст ніж іншими (рис 3.13).

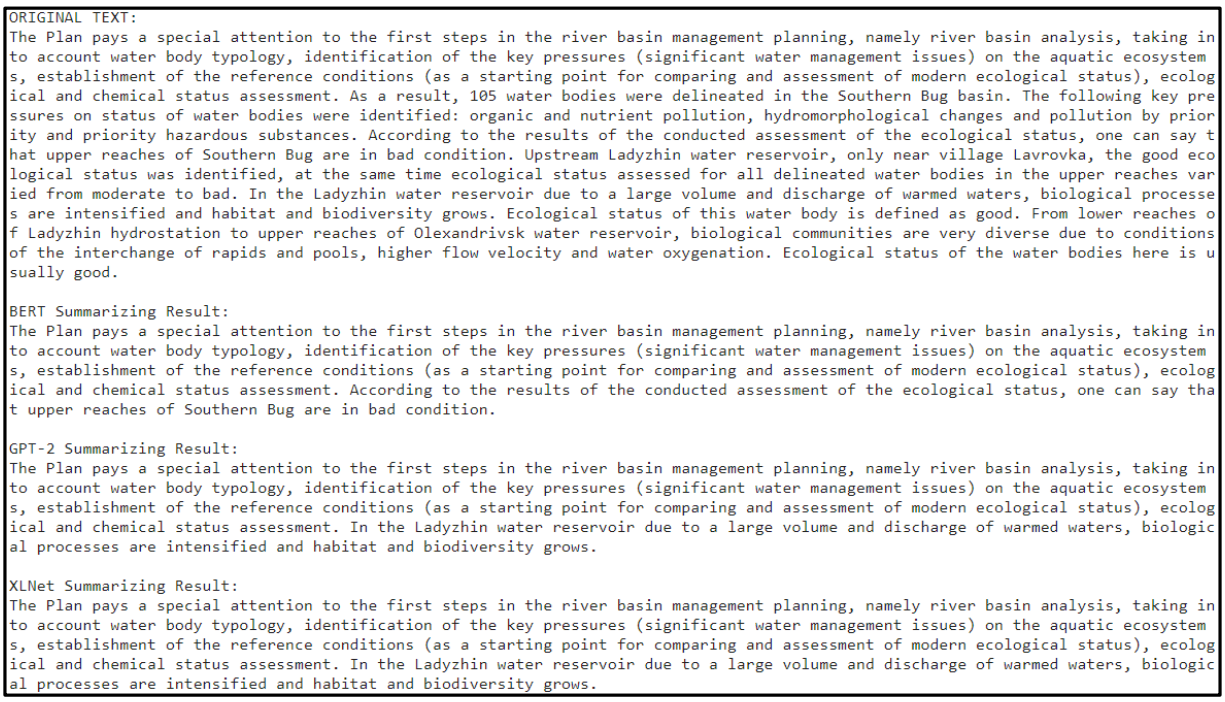

Рисунок 3.13 – Анотування тексту

Також проведемо анотування тексту з розділу 1.3. Завантажимо текст в середовище Kaggle та запустимо процес сумарайзингу (анотування) (рис. 3.14, 3.15).

```
ORIGINAL TEXT:
The degree of connectivity of the population to WWTPs varies widely - from 1
1% to 95%. The maximum rate is observed in Pivdennoukrainsk. In large cities
with EH close or greater than 100000 EH connection to sewers ranges 85-92 8
0% in towns usually no more than 30% in rural settlements. Due to lack of da
ta no accurate estimates can be given. Based on expert judgment taking that
average degree of connectivity of the urban population to WWTPs is approxima
tely 80%. The total load of inorganic compounds of nitrogen produced in 2011
at wastewater facilities with >2000 was more than 488 tons and their spatial
distribution. The obtained results show that the main part (77%) of inorgani
c nitrogen comes to waste waters from three the biggest oblast centers вЂ“ V
innitsya Khmelnitsky and Kirogovrad. The majority of the wastewater treatmen
t facilities in the Southern Bug basin are equipped with the facilities for
biological treatment of wastewaters. This method allows extracting only a pa
rt of pollutants containing nitro en and phosphorus. Technologies of deep ex
traction of these compounds are not used in Ukraine due to their high cost.
The existing wastewater treatment facilities were constructed during the Sov
iet time they are physically and technologically outdated they work ineffici
ently. Very often the volumes of wastewater exceed designed values which lea
d to additional pollution. Most of wastewater treatment facilities require u
rgent modernization and reconstruction. According to the obtained data waste
water facilities with the lowest efficiency regarding Ninorg treatment are i
n Vatutino Tulchin Bashtanka Smoline settlements. Wastewater facilities of t
he large cities ( > 100000) namely Kirovograd Vinnitsya Khmelnitsky also wor
k ineffectively and lead to increased contains of nitrogen in the surface wa
ters.
```

Рисунок 3.14 – Оригінальний текст з файлу

```
BERT Summarizing Result:
The degree of connectivity of the population to WWTPs varies widely - from 11% to 95%. Based on expe
rt judgment taking that average degree of connectivity of the urban population to WWTPs is approxima
tely 80%. Technologies of deep extraction of these compounds are not used in Ukraine due to their hi
gh cost. According to the obtained data wastewater facilities with the lowest efficiency regarding N
inorg treatment are in Vatutino Tulchin Bashtanka Smoline settlements.

GPT-2 Summarizing Result:
The degree of connectivity of the population to WWTPs varies widely - from 11% to 95%. Very often th
e volumes of wastewater exceed designed values which lead to additional pollution. Wastewater facili
ties of the large cities ( > 100000) namely Kirovograd Vinnitsya Khmelnitsky also work ineffectively
and lead to increased contains of nitrogen in the surface waters.

XLNet Summarizing Result:
The degree of connectivity of the population to WWTPs varies widely - from 11% to 95%. Based on expe
rt judgment taking that average degree of connectivity of the urban population to WWTPs is approxima
tely 80%. Very often the volumes of wastewater exceed designed values which lead to additional pollu
tion. Wastewater facilities of the large cities ( > 100000) namely Kirovograd Vinnitsya Khmelnitsky
also work ineffectively and lead to increased contains of nitrogen in the surface waters.
```

Рисунок 3.15 – Результати анотування

З даним текстом ситуація виявилась найцікавішою, тому що всі три методи анотування показали різний результат і перечитавши все, можна зробити висновки, що всі 3 моделі розпочали однаково, але далі моделі BERT та XLNet обрали за важливіше речення з думкою експертної оцінки, а модель GPT-2 речення з значенням про перевищення обсягів стічних вод, це ж саме далі показала і XLNet-модель. Закінченням анотування в моделі BERT є підсумок отриманих даних з найнижчою ефективністю, GPT-2 та XLNet обрали речення схоже за інформаційним сенсом що обрала BERT-модель, але є менш узагальнюючим. Хоча воно також має назви населених пунктів і підводить певний підсумок.

Тож можна зробити висновок, що з даним текстом анотування було проведено успішно і найкращою себе показала BERT-модель, як і з попереднім текстом. Вона обрала більш загальні речення, які, на її думку, є більш адекватними усьому тексту.

Таблиця 3.1 – Результати дослідження

Модель	Текст №1		Текст №2		Текст №3	
	Час	Результат	Час	Результат	Час	Результат
BERT	38,4 с	Гарний	39,5 с	Найкращий	42,8 с	Найкращий
GPT-2	45,9 с	Гарний	44,2 с	Гарний	61 с	Гарний
XLNet	14,7 с	Гарний	13,4 с	Гарний	18,1 с	Гарний

3.3 Висновки

В даному розділі було розроблено інформаційну технологію, побудовано та описано алгоритм роботи програми, завантажено всі необхідні бібліотеки та текст, побудовано NLP-моделі, проведено тестування системи на різних текстових даних з інформацією про водні ресурси, та визначено що їх

адекватність є однаковою, за певними винятками, детальніше можна переглянути результат побудови моделей в таблиці 3.1.

Якщо зробити загальні підсумки роботи всіх трьох моделей з трьома текстами, то найкращою по адекватності для анотування тексту є BERT-модель, вона ж є середньою по часу роботи, найшвидшою виявилась XLNet-модель, хоча її результати виявились трішки гіршими, ніж в першої, але відносно кращими ніж в GPT-2.

4 ЕКОНОМІЧНА ЧАСТИНА

4.1 Комерційний та технологічний аудит науково-технічної розробки

Метою даного розділу є проведення технологічного аудиту, в даному випадку нової інформаційної інтелектуальна технології автоматичного синтезу анотацій до матеріалів про водні ресурси. Особливістю розробки є підвищення адекватності анотацій текстовим матеріалам про водні ресурси, які синтезуються автоматично з використанням методів штучного інтелекту. Дана розробка є актуальною, тому що економія часу в 21 сторіччі є доволі актуальною проблемою, дана система дозволить робити автоматичні анотації до матеріалів про водні ресурси та економити людський час.

Аналогом може бути summary-tool, вартістю 150000 грн. Але даний аналог буде орієнтований на будь-який текст, а наша система на текст з інформацією про водні ресурси.

Для проведення комерційного та технологічного аудиту залучають не менше 3-х незалежних експертів. Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням п’ятибальної системи оцінювання за 12-ма критеріями, у відповідності із табл. 4.1 [42].

Таблиця 4.1 – Рекомендовані критерії оцінювання комерційного потенціалу розробки та їх можлива бальна оцінка

Бали (за 5-ти бальною шкалою)					
Кри те-рій	0	1	2	3	4
Технічна здійсненність концепції					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено працездат-ність продукту в реальних умовах
Ринкові переваги					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів

Продовження табл. 4.1

Бали (за 5-ти бальною шкалою)					
Кри-те рій	0	1	2	3	4
Ринкові переваги					
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в ана-логів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в ана-логів	Технічні та споживчі вла-стивості продукту значно кращі, ніж в аналогів
5	Експлуатаці йні витрати значно вищі, ніж в аналогів	Експлуатаці йні витрати дещо вищі, ніж в аналогів	Експлуатаці йні витрати на рівні експлуа-таційних ви-трат аналогів	Експлуатаці йні витрати трохи нижчі, ніж в аналогів	Експлуата ційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має пози-тивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивно ю динамі-кою

Продовження табл. 4.1

Бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
Ринкові перспективи					
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витрачати значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так із комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування

Продовження табл. 4.1

Бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
Практична здійсненність					
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років

Продовження табл. 4.1

Бали (за 5-ти бальною шкалою)					
Кри-те рій	0	1	2	3	4
Практична здійсненність					
12	Необхідна розробка регламе-нтних документів та отри-мання великої кількості дозвільних документів на ви-робництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає знач-них коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки пові-домлення відповідним органам про виро-бництво та реалізацію продукту	Відсутні будь-які регламен-тні обмеження на вироб-ництво та реалізацію продукту

Усі дані по кожному параметру занесено в таблицю 4.2.

Таблиця 4.2 – Результати оцінювання комерційного потенціалу розробки

Критерії оцінювання	ПІБ експертів		
	Лопухов Б.Л.	Пасічнюк Д.В.	Радецький О.В.
	Бали, виставлені експертами:		
Технічна здійсненність концепції	4	4	4
Наявність аналогів на ринку	4	4	4
Цінова політика	4	4	4
Технічні та споживчі властивості виробу	4	3	4
Експлуатаційні витрати	4	4	3
Ринок збуту	4	3	4
Конкурентоспроможність	3	4	3
Фахівці з технічної і комерційної реалізації	3	3	2
Фінансування	4	4	3
Матеріально-технічна база	3	3	3
Термін реалізації ідеї	4	4	4
Супровідна документація	2	3	4
Сума	43	43	42
Середньоарифметична сума балів	(43+43+42) / 3 = 42,67		

За даними таблиці 4.2 можна зробити висновок щодо рівня комерційного потенціалу даної розробки. Для цього доцільно скористатись рекомендаціями, наведеними в таблиці 4.3 [42].

Таблиця 4.3 - Рівні комерційного потенціалу розробки

Середньоарифметична сума балів, розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0-10	Низький
11-20	Нижче середнього
21-30	Середній
31-40	Вище середнього
41-48	Високий

Як видно з таблиці, рівень комерційного потенціалу розроблюваного нового програмного продукту є високим, що досягається за рахунок підвищення адекватності анотацій текстовим матеріалам про водні ресурси, які синтезуються автоматично з використанням методів штучного інтелекту. Дана розробка є актуальною, тому що економія часу в 21 сторіччі є доволі актуальною проблемою, дана система дозволить робити автоматичні анотації до матеріалів про водні ресурси так економити людський час.

Наукова новизна полягає в тому, що дістала подальший розвиток інтелектуальна інформаційна технологія автоматичного синтезу анотацій до матеріалів про водні ресурси за рахунок підвищення адекватності цих анотацій.

4.2 Прогнозування витрат на виконання науково-дослідної (дослідно-конструкторської) роботи

1. Основна заробітна плата розробників.

Основна заробітна плата розраховується за формулою:

$$З_о = \frac{М}{Т_р} \cdot t, \quad (4.1)$$

де М – місячний посадовий оклад конкретного розробника (дослідника), грн;

$Т_р$ – число робочих днів за місяць, 21 днів;

t – число днів роботи розробника (дослідника).

Результати розрахунків зведемо до таблиці 4.4.

Таблиця 4.4 – Основна заробітна плата розробників

Найменування посади	Місячний посадовий оклад, грн.	Оплата за робочий день, грн.	Число днів роботи	Витрати на заробітну плату, грн.
Керівник проекту	41000	1952,38	33	64428,571
Програміст	35000	1666,67	33	55000,000
Всього				119428,57

Так як в даному випадку розробляється програмний продукт, то розробник виступає одночасно і основним робітником, і тестувальником розроблюваного програмного продукту.

2. Розрахунок додаткової заробітної плати розробників.

Додаткову заробітну плату прийнято розраховувати як 14 % від основної заробітної плати розробників та робітників:

$$З_д = З_о \cdot 14\ \% / 100\ \% \quad (4.2)$$

$$З_д = (119428{,}57 \cdot 14\ \% / 100\ \%) = 16720{,}00\ (грн).$$

Згідно діючого законодавства нарахування на заробітну плату складають 22 % від суми основної та додаткової заробітної плати:

$$Н_з = (З_о + З_д) \cdot 22\ \% / 100\% \quad (4.3)$$

$$Н_з = (119428{,}57 + 16720{,}00) \cdot 22\ \% / 100\ \% = 29952{,}69\ (грн).$$

Оскільки для розроблювального пристрою не потрібно витрачати матеріали та комплектуючи, то витрати на матеріали і комплектуючі дорівнюють нулю.

3. Амортизація обладнання, що використовувалось для розробки в спрощеному вигляді розраховується за формулою:

$$А = \frac{Ц}{Тв} \cdot \frac{t_{вик}}{12}\ [грн]. \quad (4.4)$$

де Ц – балансова вартість обладнання, грн.;

Т – термін корисного використання обладнання згідно податкового законодавства, років;

$t_{вик}$ – термін використання під час розробки, місяців.

Розрахуємо, для прикладу, амортизаційні витрати на комп’ютер балансова вартість якого становить 28000 (грн), термін його корисного використання згідно податкового законодавства – 2 роки, а термін його фактичного використання – 1,57 міс:

$$А_{обл} = \frac{28000}{2} \times \frac{1{,}57}{12} = 1833{,}33\ (грн). \quad (4.5)$$

Аналогічно визначаємо амортизаційні витрати на інше обладнання та приміщення. Розрахунки заносимо до таблиці 4.5.

Так як вартість ліцензійної ОС та спеціалізованих ліцензійних нематеріальних активів є безкоштовною, то $В_{нем.ак.} = 0$ (грн).

Таблиця 4.5 – Амортизаційні відрахування на матеріальні та нематеріальні ресурси для розробників

Найменування обладнання	Балансова вартість, грн.	Строк корисного використання, років	Термін використання обладнання, місяців	Амортиза-ційні відрахування-ня, грн.
Комп'ютер та комп'ютерна периферія **(**Ноутбук HP Pavilion Gaming 15-cx0033ua**)**	28000	2	1,57	1833,333
Офісне обладнання (меблі)	23000	4	1,57	752,976
Приміщення	880000	20	1,57	5761,905
Всього				8348,21

Тарифи на електроенергію для непобутових споживачів (промислових підприємств) відрізняються від тарифів на електроенергію для населення. При цьому тарифи на розподіл електроенергії у різних постачальників (енергорозподільних компаній), будуть різними. Крім того, розмір тарифу залежить від класу напруги (1-й або 2-й клас). Тарифи на розподіл електроенергії для всіх енергорозподільних компаній встановлює Національна комісія з регулювання енергетики і комунальних послуг (НКРЕКП). Витрати на силову електроенергію розраховуються за формулою:

$$\tau = d + f, , \tag{4.5}$$

де В – вартість 1 кВт-години електроенергії для 1 класу підприємства, В = 6,2 грн./кВт;

П – встановлена потужність обладнання, кВт. П = 0,45 кВт;

Ф – фактична кількість годин роботи обладнання, годин;

$К_п$ – коефіцієнт використання потужності, $К_п$ = 0,9.

$$В_е = 0,9 \cdot 0,45 \cdot 8 \cdot 33 \cdot 6,2 = 662,904 \text{ (грн)}.$$

4. Інші витрати та загальновиробничі витрати

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками. Витрати за статтею «Інші витрати» розраховуються як 50…100% від суми основної заробітної плати дослідників:

$$І_в = (З_о + З_р) \cdot \frac{Н_{ів}}{100\%}, \tag{4.6}$$

де $Н_{ів}$ – норма нарахування за статтею «Інші витрати».

$$І_в = 119428,57 * 80\% / 100\% = 95542,86 \text{ (грн)}.$$

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов’язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов’язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов’язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін. Витрати за статтею «Накладні

(загальновиробничі) витрати» розраховуються як 100…150% від суми основної заробітної плати дослідників:

$$Н_{нзв} = (З_о + З_р) \cdot \frac{Н_{нзв}}{100\%}, \quad (4.7)$$

де $Н_{нзв}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати».

$$Н_{нзв} = 119428,57 * 110\ \% / 100\ \% = 131371 \text{ (грн)}.$$

5. Витрати на проведення науково-дослідної роботи та загальні витрати

Сума всіх попередніх статей витрат дає загальні витрати на проведення науково-дослідної роботи:

$$В_{заг} = 119428,57+16720,00+29952,69+8348,21+662,90+95542,86+$$
$$+131371 = 402026,66 \text{ (грн)}.$$

Розрахунок загальних витрат на науково-дослідну роботу (науково-технічну) роботу та оформлення її результатів.

Загальні витрати на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховуються за формулою:

$$ЗВ = \frac{В_{заг}}{\eta} \text{ [грн]}, \quad (4.8)$$

де η – коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи.

Так, якщо науково-технічна розробка знаходиться на стадії: науково-дослідних робіт, то η=0,1; технічного проектування, то η=0,2; розробки

конструкторської документації, то η=0,3; розробки технологій, то η=0,4; розробки дослідного зразка, то η=0,5; розробки промислового зразка, то η=0,7; впровадження, то η=0,9. Оберемо η = 0,5, так як розробка, на даний момент, знаходиться на стадії дослідного зразка:

$$ЗВ = 402026,66 / 0,5 = 804053 \text{ (грн)}.$$

4.3 Розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором

В ринкових умовах узагальнювальним позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів тієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку. Саме зростання чистого прибутку забезпечить потенційному інвестору надходження додаткових коштів, дозволить покращити фінансові результати його діяльності, підвищить конкурентоспроможність та може позитивно вплинути на ухвалення рішення щодо комерціалізації цієї розробки.

Для того, щоб розрахувати можливе зростання чистого прибутку у потенційного інвестора від можливого впровадження науково-технічної розробки необхідно:

а) вказати, з якого часу можуть бути впроваджені результати науково-технічної розробки;

б) зазначити, протягом скількох років після впровадження цієї науково-технічної розробки очікуються основні позитивні результати для потенційного інвестора (наприклад, протягом 3-х років після її впровадження);

в) кількісно оцінити величину існуючого та майбутнього попиту на цю або аналогічні чи подібні науково-технічні розробки та назвати основних суб’єктів (зацікавлених осіб) цього попиту;

г) визначити ціну реалізації на ринку науково-технічних розробок з аналогічними чи подібними функціями.

При розрахунку економічної ефективності потрібно обов'язково враховувати зміну вартості грошей у часі, оскільки від вкладення інвестицій до отримання прибутку минає чимало часу. При оцінюванні ефективності інноваційних проектів передбачається розрахунок таких важливих показників:

– абсолютного економічного ефекту (чистого дисконтованого доходу);

– внутрішньої економічної дохідності (внутрішньої норми дохідності);

– терміну окупності (дисконтованого терміну окупності).

Аналізуючи напрямки проведення науково-технічних розробок, розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором можна об'єднати, враховуючи визначені ситуації з відповідними умовами.

1. Розробка чи суттєве вдосконалення програмного забезпечення.

В цьому випадку майбутній економічний ефект буде формуватися на основі таких даних:

$$\Delta П_i = (\pm \Delta Ц_0 \cdot N + Ц_0 \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot (1 - \frac{\vartheta}{100}), \qquad (4.9)$$

де $\pm\Delta Ц_о$ – зміна вартості програмного продукту (зростання чи зниження) від впровадження результатів науково-технічної розробки в аналізовані періоди часу;

N – кількість споживачів які використовували аналогічний продукт у році до впровадження результатів нової науково-технічної розробки;

$Ц_о$ – основний оціночний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки, $Ц_о = Ц_б \pm \Delta Ц_о$;

Цб – вартість програмного продукту у році до впровадження результатів розробки;

ΔN – збільшення кількості споживачів продукту, в аналізовані періоди часу, від покращення його певних характеристик;

λ – коефіцієнт, який враховує сплату податку на додану вартість. Ставка податку на додану вартість дорівнює 20%, а коефіцієнт λ = 0,8333.

р – коефіцієнт, який враховує рентабельність продукту;

ϑ – ставка податку на прибуток, у 2022 році ϑ =18%.

Припустимо, що при прогнозованій ціні 5500 грн. за одиницю виробу, термін збільшення прибутку складе 3 роки. Після завершення розробки і її вдосконалення, можна буде підняти її ціну на 550 грн. Кількість одиниць реалізованої продукції також збільшиться: протягом першого року – на 2200 шт., протягом другого року – на 1500 шт., протягом третього року на 1000 шт. До моменту впровадження результатів наукової розробки реалізації продукту не було:

$\Delta П_1$ = (0*550 + (5500 + 550)*2200)* 0,8333* 0,38) * (1 - 0,18) = 3141966,541 (грн).

$\Delta П_2$ = (0*550 + (5500 + 550)*(2200+1500)* 0,8333* 0,38) * (1 - 0,18) = 5812638,101 (грн).

$\Delta П_3$ = (0*550 + (5500 + 550)*(2200+1500+1000)* 0,8333* 0,38) * (1 - 0,18) = 7383621,371 (грн).

Отже, комерційний ефект від реалізації результатів розробки за три роки складе 16338226,01 грн.

2. Розрахунок ефективності вкладених інвестицій та періоду їх окупності

Розраховуємо приведену вартість збільшення всіх чистих прибутків *ПП*, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_1^T \frac{\Delta П_i}{(1+\tau)^t}, \quad (4.10)$$

де $\Delta П_i$ –збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої науково-дослідної (науково-технічної) роботи, грн;

Т – період часу, протягом якою виявляються результати впровадженої науково-дослідної (науково-технічної) роботи, роки;

τ –ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau = 0,05…0,15$;

t – період часу (в роках).

Збільшення прибутку ми отримаємо, починаючи з першого року:

$$ПП = (3141966,541/(1+0,1)^1)+(5812638,101/(1+0,1)^2)+(7383621,371/$$
$$/(1+0,1)^3) = 2856333,22 + 4803833,141 + 5547424,021 = 13207590,38 \text{ (грн).}$$

Далі розраховують величину початкових інвестицій *PV*, які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки. Для цього можна використати формулу:

$$PV = k_{інв} * ЗВ, \quad (4.11)$$

де $k_{інв}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію. Це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо; зазвичай $k_{інв}$=2...5, але може бути і більшим;

ЗВ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, грн.

$$PV = 2 * 804053 = 1608106,64 \text{ (грн).}$$

Тоді абсолютний економічний ефект Еабс або чистий приведений дохід (NPV, Net Present Value) для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$Е_{абс} = ПП – PV, \quad (4.12)$$

$$Е_{абс} = 13207590,38 – 1608106,64 = 11599483,74 \text{ (грн)}.$$

Оскільки $Е_{абс} > 0$то вкладання коштів на виконання та впровадження результатів даної науково-дослідної (науково-технічної) роботи може бути доцільним.

Для остаточного прийняття рішення з цього питання необхідно розрахувати внутрішню економічну дохідність або показник внутрішньої но-рми дохідності (IRR, Internal Rate of Return) вкладених інвестицій та порівняти її з так званою бар'єрною ставкою дисконтування, яка визначає ту мінімальну внутрішню економічну дохідність, нижче якої інвестиції в будь-яку науково-технічну розробку вкладати буде економічно недоцільно.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій $Е_в$. Для цього використаємо формулу:

$$E_в = \sqrt[T_ж]{1 + \frac{E_{абс}}{PV}} - 1, \quad (4.13)$$

$Т_ж$ –життєвий цикл наукової розробки, роки.

$$Е_в = \sqrt[3]{(1 + 11599483,74/1608106,64} - 1 = 1,018$$

Визначимо мінімальну ставку дисконтування, яка у загальному вигляді визначається за формулою:

$$\tau = d + f, \tag{4.14}$$

де d – середньозважена ставка за депозитними операціями в комерційних банках в 2022 році в Україні d = (0,09…0,14);

f – показник, що характеризує ризикованість вкладень; зазвичай, величина f = (0,05...0,5).

$$\tau_{min.} = 0{,}14 + 0{,}05 = 0{,}19$$

Так як $E_в > \tau_{min}$, то інвестор може бути зацікавлений у фінансуванні даної наукової розробки.

Розрахуємо термін окупності вкладених у реалізацію наукового проекту інвестицій за формулою:

$$T_{ок} = \frac{1}{E_в}, \tag{4.15}$$

$$T_{ок} = 1 / 1{,}018 = 0{,}98 \text{ р.}$$

Оскільки $T_{ок}$ < 3-х років, а саме термін окупності рівний 0,98 років, то фінансування даної наукової розробки є доцільним.

4.4 Висновки

Економічна частина даної роботи містить розрахунок витрат на розробку нового програмного продукту, сума яких складає 804053 гривень. Було спрогнозовано орієнтовану величину витрат по кожній з статей витрат. Також розраховано чистий прибуток, який може отримати виробник від реалізації нового технічного рішення, розраховано період окупності витрат для інвестора та економічний ефект при використанні даної розробки. В

результаті аналізу розрахунків можна зробити висновок, що розроблений програмний продукт за ціною дешевший за аналог і є висококонкурентоспроможним. Період окупності складе близько 0,98 років.

ВИСНОВКИ

В ході магістерської кваліфікаційної роботи проведено аналіз предметної області, проведено розвідувальний аналіз, обрано та побудовано оптимальні NLP-моделі.

Проблема автоматичного синтезу тексту є доволі серйозною проблемою для людини, тому що у XXI сторіччі люди роблять все доволі швидко та шукають способи економії часу різними методами. Проблема якості водних ресурсів є також однією з найважливіших в світі, на цю тему є безліч інформації, досліджень, статей. І для того, щоб зекономити людський час, було розроблено інформаційну інтелектуальну технологію автоматичного синтезу анотацій до матеріалів про водні ресурси, яка показує гарний результат з різним текстом.

Визначено суть наявної проблеми, розглянуто існуючі методи вирішення даної проблеми, аналог, а саме – проведено тестування синтезу тексту методами штучного інтелекту, розглянуто бібліотеки, які будуть використовуватись у роботі, розглянуто проблему анотування тексту в ГІС-технологіях та їх геоприв’язування чи геокодування.

Проведено огляд датасету з англомовним текстом, здійснено розвідувальний аналіз по параметру «Забруднення», а саме – побудовано різні графіки по різним критеріям. Проведено аналіз сучасних мовних моделей і технологій штучного інтелекту і як оптимальні вибрано моделі BERT, GPT-2, XLNet. Також були обрані бібліотеки «Bert Extractive Summarizer», «Transformers», «spaCy» для проведення адекватного анотування.

Було розроблено інформаційну технологію, побудовано та описано алгоритм роботи програми, завантажено всі необхідні бібліотеки та текст, побудовано NLP-моделі, проведено тестування системи на різних текстових даних з інформацією про водні ресурси, та визначено, що їх адекватність є однаковою, за певними винятками.

Якщо зробити загальні підсумки роботи всіх трьох моделей з трьома текстами, то найкращою по адекватності для анотування тексту є BERT-модель, вона ж є середньою по часу роботи, найшвидшою виявилась XLNet-модель, хоча її результати виявились трішки гіршими, ніж в першої, але відносно кращими ніж в GPT-2.

Економічна частина даної роботи містить розрахунок витрат на розробку нового програмного продукту, сума яких складає 804053 гривень. Було спрогнозовано орієнтовану величину витрат по кожній з статей витрат. Також розраховано чистий прибуток, який може отримати виробник від реалізації нового технічного рішення, розраховано період окупності витрат для інвестора та економічний ефект при використанні даної розробки. В результаті аналізу розрахунків можна зробити висновок, що розроблений програмний продукт за ціною дешевший за аналог і є висококонкурентоспроможним. Період окупності складе близько 0,98 років.

За результатами даної роботи помітне підвищення адекватності анотацій текстовим матеріалам про водні ресурси, які синтезуються автоматично з використанням методів штучного інтелекту. Отже, завдання, поставлене у магістерській кваліфікаційній роботі, виконано в повному обсязі.

За результатами даної роботи написані тези доповіді, які були апробовані на Всеукраїнській науково-практичній інтернет-конференції «Молодь в науці: дослідження, проблеми, перспективи» (Вінниця, 2021-2022 рр.) та опубліковані у матеріалах цієї конференції.

Дана магістерська кваліфікаційна робота виконана на замовлення басейнової ради Південного Бугу, куди й передано її результати. Про них вже йшлось в обговоренні до п.4 порядку денного засідання ради (див. протокол № 12 від 07.12.2022 р. на сайті БУВР Південного Бугу).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Мокін В. Б, Крижановський Є. М., Гораш, М. А., Пасічнюк Д. В., Радецький О. В., Хмарук С. М. Інформаційні технології збирання та систематизування даних про стан та проблеми масивів вод на прикладі басейну річки Південний Буг. *Молодь в науці: дослідження, проблеми, перспективи (МН-2022)*. Вінниця, 2022. URL: https://conferences.vntu.edu.ua/index.php/mn/mn2022/paper/viewFile/16236/13677

2. Савченко А.С., Синельніков О.О. Методи та системи штучного інтелекту: навч. посіб. Київ : ЦУЛ, 2021. 9 с.

3. TEXT SUMMARIZATION IN NLP. URL: https://www.topcoder.com/thrive/articles/text-summarization-in-nlp

4. Грицай Д.І. Розумні роботи, *Вісник ЛНУ*. 2020. № 1/2. С. 104

5. Мокін В.Б.. NLP for WR : Summarizing using BERT, GPT2, XLNET, 2022. URL: https://www.kaggle.com/code/vbmokin/nlp-for-wr-summarizing-using-bert-gpt2-xlnet

6. NLP і модель «мішка слів». URL: https://dou.ua/lenta/articles/first-steps-in-nlp-nltk/

7. Types of Text Summarization: Extractive and Abstractive Summarization Basics. URL: https://turbolab.in/types-of-text-summarization-extractive-and-abstractive-summarization-basics/

8. Text Summarization. URL: https://devopedia.org/text-summarization

9. NLP-технології розпізнавання людської мови. Можливості та сфери застосування. URL: https://evergreens.com.ua/ru/articles/natural-language-processing.html

10. Резюмування тексту для NLP: 5 найкращих API у 2023 році. URL: https://www.assemblyai.com/blog/text-summarization-nlp-5-best-apis/

11. Seaborn: statistical data visualization URL: https://seaborn.pydata.org/

12. Бібліотека Numpy. URL: https://pythobyte.com/numpy-tutorial-a-simple-example-based-guide-ff3127f2/

13. Бібліотека Pandas. URL: https://pandas.pydata.org/docs/getting_started/overview.html

14. Бібліотека Matplotlib. URL: https://matplotlib.org/

15. Бібліотека Summarizer. URL: https://www.activestate.com/blog/how-to-do-text-summarization-with-python/

16. Бібліотека Scikit-Learn. URL: https://neurohive.io/ru/osnovy-data-science/vvedenie-v-scikit-learn/

17. Бібліотека Warnings URL: https://docs.python.org/3/library/warnings.html

18. Мокін В. Б., Крижановський Є. М. Геоінформаційні системи в екології: навч. посіб / за ред. Є. М. Крижановського. Вінниця : ВНТУ, 2014. 61 с.

19. Мокін В. Б., Собко Б. Ю., Крижановський Є. М., Дратований М.В., Горячев Г. В.. Створення інформаційної системи моніторингу забруднення атмосферного повітря міста на основі технології «Інтернет речей», *Вісник ВПІ*. 2017. № 3. С. 49-58.

20. Визначення масивів поверхневих та підземних вод. URL: https://poltavavodgosp.gov.ua/vyznachennia-masyviv-poverkhnevykh-ta-pidzemnykh-vod/

21. Продукти ArcGIS Desktop URL: http://www.geoguide.com.ua/software/software.php?part=esri&art=esri

22. GIS Map Viewer. URL: https://linkgis.org/mapviewer/

23. Digitals. URL: http://www.geoguide.com.ua/software/software.php?part=geosystema&art=digitals

24. Visicom. URL: https://api.visicom.ua/

25. Мокін В. Б.. NLP : Report & News Classification. URL: https://www.kaggle.com/datasets/vbmokin/nlp-reports-news-classification

26. План управління річковим басейном Південного Бугу: аналіз стану та заходи / Мокін В. Б. та ін. ; за ред. Афанасьєва С., Пітерса А., Сташука А., Ярошевича О. Київ : Знання, 2014. 188 с.

27. Хмарук С. М. ER-Pollution - NLP : Bag of Words, TF IDF, GloVe. URL: https://www.kaggle.com/code/khmaruk/er-pollution-nlp-bag-of-words-tf-idf-glove

28. Гупта С. К.. Реферативне узагальнення: огляд сучасного стану, 2020, Київ : НБУВ, С. 49-65

29. Адхікарі С. Підходи машинного навчання для резюмування тексту на основі NLP. *Четверта міжнародна конференція з обчислювальних методологій і комунікацій (ICCMC)*. м. Ерод, 11–13 березня 2020 р. С. 2-4.

30. BERT (Language Model). URL: https://devopedia.org/bert-language-model

31. 10 Leading Language Models For NLP In 2022. URL: https://www.topbots.com/leading-nlp-language-models-2020/#language-models-2022-10

32. GPT-2 (GPT2) vs. GPT-3 (GPT3): The OpenAI Showdown. URL: https://dzone.com/articles/gpt-2-gpt2-vs-gpt-3-gpt3-the-openai-showdown

33. XLNet Outperforms BERT on Several NLP Tasks. URL: https://www.kdnuggets.com/2019/07/xlnet-outperforms-bert-several-nlp-tasks.html

34. BERT for text summarization. URL: https://iq.opengenus.org/bert-for-text-summarization/

35. Transformers Language Interpretability. URL: https://www.analyticsvidhya.com/blog/2020/03/spacy-tutorial-learn-natural-language-processing/

36. spaCy Tutorial to Learn and Master Natural Language Processing (NLP). URL: https://www.analyticsvidhya.com/blog/2020/03/spacy-tutorial-learn-natural-language-processing/

37. Natural Language Toolkit. URL: https://www.nltk.org/

38. NLTK Python Tutorial (Natural Language Toolkit). URL: https://data-flair.training/blogs/nltk-python-tutorial/

39. Why Gensim?. URL: https://radimrehurek.com/gensim/

40. Generating Word Embeddings with Gensim's word2vec. URL: https://sematext.com/blog/word-embeddings-gensim-word2vec-tutorial/

41. Хмарук С. М. NLP for WR : Summarizing using BERT, GPT2, XLNET. URL: https://www.kaggle.com/code/khmaruk/nlp-for-wr-summarizing-using-bert-gpt2-xlnet/notebook

42. Козловський В. О., Лесько О. Й., Кавецький В. В. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт : уклад. Вінниця : ВНТУ, 2021. 42 с.

Додаток А

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації

ЗАТВЕРДЖУЮ
Завідувач кафедри САІТ
_____ д.т.н., проф. Мокін В. Б.
«_19_» _____09_____ 2022 р.

ТЕХНІЧНЕ ЗАВДАННЯ
на магістерську кваліфікаційну роботу
«ІНФОРМАЦІЙНА ІНТЕЛЕКТУАЛЬНА ТЕХНОЛОГІЯ АВТОМАТИЧНОГО СИНТЕЗУ АНОТАЦІЙ ДО МАТЕРІАЛІВ ПРО ВОДНІ РЕСУРСИ»
08-53.МКР.011.02.000.ТЗ

Керівник: д.т.н., проф. каф. САІТ
_____ Мокін В. Б.
«_19_» _____09_____ 2022 р.

Розробив: студент гр. 2ІСТ-21м
_____ Хмарук С. М.
«_19_» _____09_____ 2022 р.

Вінниця 2022

1. Підстава для проведення робіт

Підставою для виконання роботи є наказ № 203 по ВНТУ від «14» _09_ 2022 р., та індивідуальне завдання на МКР, затверджене протоколом № 3 засідання кафедри САІТ від «14» ___09____ 2022 р.

2. Джерела розробки:

- План управління річковим басейном Південного Бугу: аналіз і заходи річкового басейну (англ.) / Мокін В. Б. та ін. ; за ред. С. Афанасьєв, А. Петерс, В. Сташук, О. Ярошевич., Київ, 2014. 188 с.
- «NLP : Reports & News Classification» ENG & UKR Automatic Environmental Reports & News Classification. URL: https://www.kaggle.com/datasets/vbmokin/nlp-reports-news-classification

3. Мета і призначення роботи:

Підвищення адекватності анотацій текстовим матеріалам про водні ресурси, які синтезуються автоматично з використанням методів штучного інтелекту.

4. Вихідні дані для проведення робіт:

Датасет Kaggle «NLP : Reports & News Classification» з даними про водні ресурси.

5.Методи дослідження:

- розвідувальний аналіз;
- методи NLP.

6. Етапи роботи і терміни їх виконання:

1. Аналіз предметної області ..... 20.09 – 30.09
2. Вибір оптимальних технологій та проведення розвідувального аналізу..... 01.10 – 20.10
3. Розроблення інтелектуальної інформаційної технології .. 21.10 – 05.11
4. Економічна частина. ..... 06.11 – 19.11
5. Оформлення матеріалів до захисту МКР. ..... 20.11 – 30.11

7. Очікувані результати та порядок реалізації:

Адекватне визначення анотацій текстовим матеріалам про водні ресурси, які синтезуються автоматично з використанням методів штучного інтелекту

8. Вимоги до розробленої документації

Пояснювальна записка оформлена у відповідності до вимог «Методичних вказівок до виконання та оформлення магістерських кваліфікаційних робіт для студентів спеціальності 126 – «Інформаційні системи та технології» денної форми навчання».

9. Порядок приймання роботи

Публічний захист ..... «_19_» _____12_____ 2022 р.

Початок розробки..... «_20_» _____09_____ 2022 р.

Граничні терміни виконання МКР ..... «_30_» __ ____11____ _ 2022 р.

Розробив студент групи 2ІСТ-21м _____ Хмарук С. М.

Додаток Б

Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень

Назва роботи: «Інформаційна інтелектуальна технологія автоматичного синтезу анотацій до матеріалів про водні ресурси»
Тип роботи: магістерська кваліфікаційна робота
Підрозділ: кафедра САІТ
Науковий керівник: Мокін В. Б. д.т.н., проф. каф. САІТ

Показники звіту подібності Unicheck

Оригінальність	96,6 %
Схожість	3,4 %

Аналіз звіту подібності (відмітити потрібне):

✓ Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.

☐ Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і самостійності її автора. Роботу направити на розгляд експертної комісії кафедри.

☐ Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Опис прийнятого рішення:

Робота допускається до захисту

Особа, відповідальна за перевірку Жуков С. О.

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи Хмарук С. М.

Керівник роботи Мокін В. Б

Додаток В

Лістинг програми

```
%%time
!pip install bert-extractive-summarizer==0.7.1
%%time
!pip install transformers==2.2.0
%%time
!pip install spacy==2.0.12
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
from summarizer import Summarizer,TransformerSummarizer
from sklearn.metrics import accuracy_score
import warnings
warnings.filterwarnings('ignore')
df = pd.read_csv('../input/nlp-reports-news-classification/water_problem_nlp_en_for_Kaggle_100.csv', delimiter=';', header=0)
df = df.fillna(0)
convert_dict = {'text': str,
                'env_problems': int,
                'pollution': int,
                'treatment': int,
                'climate': int,
                'biomonitoring': int}
df = df.astype(convert_dict)
df = df[:5]
df
```

```
df.info()
df['text'].head(10)
df['text'].str.len().max()
max_length = 400
i = 0
bodies = []
while i < len(df):
    body = ""
    body_empty = True
    while (len(body) < max_length) and (i < len(df)):
        if body_empty:
            body = df.loc[i,'text']
            body_empty = False
        else: body += " " + df.loc[i,'text']
        i += 1
    bodies.append(body)
    print("Length of blocks =", len(body))
print(f"\nNumber of text blocks = {len(bodies)}\n")
print("Text blocks:\n", bodies)
tweet.head(10)
Real_len = tweet[tweet['env_problems'] == 1].shape[0]
Not_len = tweet[tweet['env_problems'] == 0].shape[0]
plt.rcParams['figure.figsize'] = (7, 5)
plt.bar(10,Real_len,3, label="Real", color='blue')
plt.bar(15,Not_len,3, label="Not", color='red')
plt.legend()
plt.ylabel('Number of examples')
plt.title('Propertion of examples')
plt.show()
tweet['length'] = tweet['text'].apply(length)
```

```
plt.rcParams['figure.figsize'] = (10.0, 4.0)
bins = 150
plt.hist(tweet[tweet['env_problems'] == 0]['length'], alpha = 0.6, bins=bins, label='Not')
plt.hist(tweet[tweet['env_problems'] == 1]['length'], alpha = 0.8, bins=bins, label='Real')
plt.xlabel('length')
plt.ylabel('numbers')
plt.legend(loc='upper right')
plt.xlim(0,150)
plt.grid()
plt.show()
fig,(ax1,ax2)=plt.subplots(1,2,figsize=(10,5))
tweet_len=tweet[tweet['env_problems']==1]['text'].str.len()
ax1.hist(tweet_len,color='blue')
ax1.set_title('disaster tweets')
tweet_len=tweet[tweet['env_problems']==0]['text'].str.len()
ax2.hist(tweet_len,color='red')
ax2.set_title('Not disaster tweets')
fig.suptitle('Characters in tweets')
plt.show()
fig,(ax1,ax2)=plt.subplots(1,2,figsize=(10,5))
tweet_len=tweet[tweet['env_problems']==1]['text'].str.split().map(lambda x: len(x))
ax1.hist(tweet_len,color='blue')
ax1.set_title('Disaster tweets')
tweet_len=tweet[tweet['env_problems']==0]['text'].str.split().map(lambda x: len(x))
ax2.hist(tweet_len,color='red')
ax2.set_title('Not disaster tweets')
fig.suptitle('Words in a tweet')
plt.show()
```

```
fig,(ax1,ax2)=plt.subplots(1,2,figsize=(10,5))
word=tweet[tweet['env_problems']==1]['text'].str.split().apply(lambda x : [len(i) for i in x])
sns.distplot(word.map(lambda x: np.mean(x)),ax=ax1,color='blue')
ax1.set_title('Disaster')
word=tweet[tweet['env_problems']==0]['text'].str.split().apply(lambda x : [len(i) for i in x])
sns.distplot(word.map(lambda x: np.mean(x)),ax=ax2,color='red')
ax2.set_title('Not disaster')
fig.suptitle('Average word length in each tweet')
min_length_text = 40
%%time
bert_summary = []
for i in range(len(bodies)):
    bert_model = Summarizer()
    bert_summary.append(''.join(bert_model(bodies[i],
min_length=min_length_text)))
%%time
gpt_summary = []
for i in range(len(bodies)):
    GPT2_model =
TransformerSummarizer(transformer_type="GPT2",transformer_model_key="gpt2-medium")
    gpt_summary.append(''.join(GPT2_model(bodies[i],
min_length=min_length_text)))
%%time
xlnet_summary = []
for i in range(len(bodies)):
```

```
    model = TransformerSummarizer(transformer_type="XLNet",transformer_model_key="xlnet-base-cased")
    xlnet_summary.append(''.join(model(bodies[i], min_length=min_length_text)))
%%time
print("All Summarizing Results:\n")
for i in range(len(bodies)):
    print("ORIGINAL TEXT:")
    print(bodies[i])
    print("\nBERT Summarizing Result:")
    print(bert_summary[i])
    print("\nGPT-2 Summarizing Result:")
    print(gpt_summary[i])
    print("\nXLNet Summarizing Result:")
    print(xlnet_summary[i])
    print("\n\n")
```

Додаток Г

ІЛЮСТРАТИВНА ЧАСТИНА

ІНФОРМАЦІЙНА ІНТЕЛЕКТУАЛЬНА ТЕХНОЛОГІЯ АВТОМАТИЧНОГО СИНТЕЗУ АНОТАЦІЙ ДО МАТЕРІАЛІВ ПРО ВОДНІ РЕСУРСИ

Виконав: студент гр. 2ІСТ-21м
_____ Хмарук С. М.
«_01_» _____12_____ 2022 р.

Керівник: д.т.н., проф. каф. САІТ
_____ Мокін В. Б.
«_02_» _____12__ _____ 2022 р.

Нормоконтроль: к.т.н., доцент
_____ Жуков С. О.
«_02_» _____12_____ 2022 р.

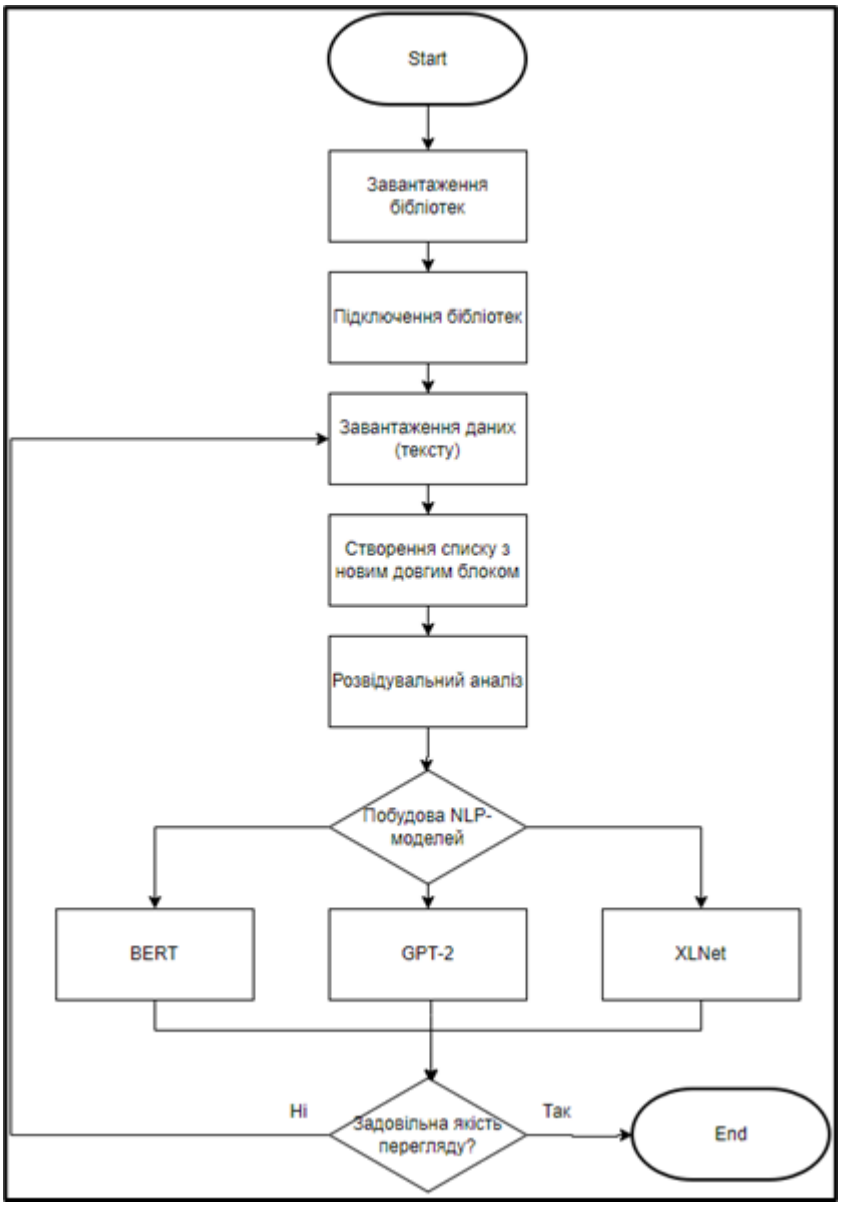

Рисунок Г.1 – Алгоритм роботи програми

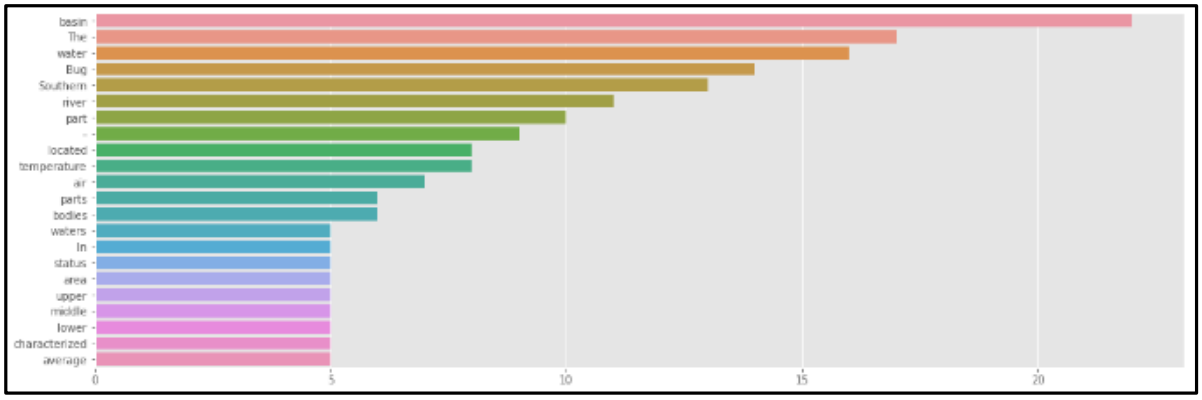

Рисунок Г.2 – Загальновживані слова

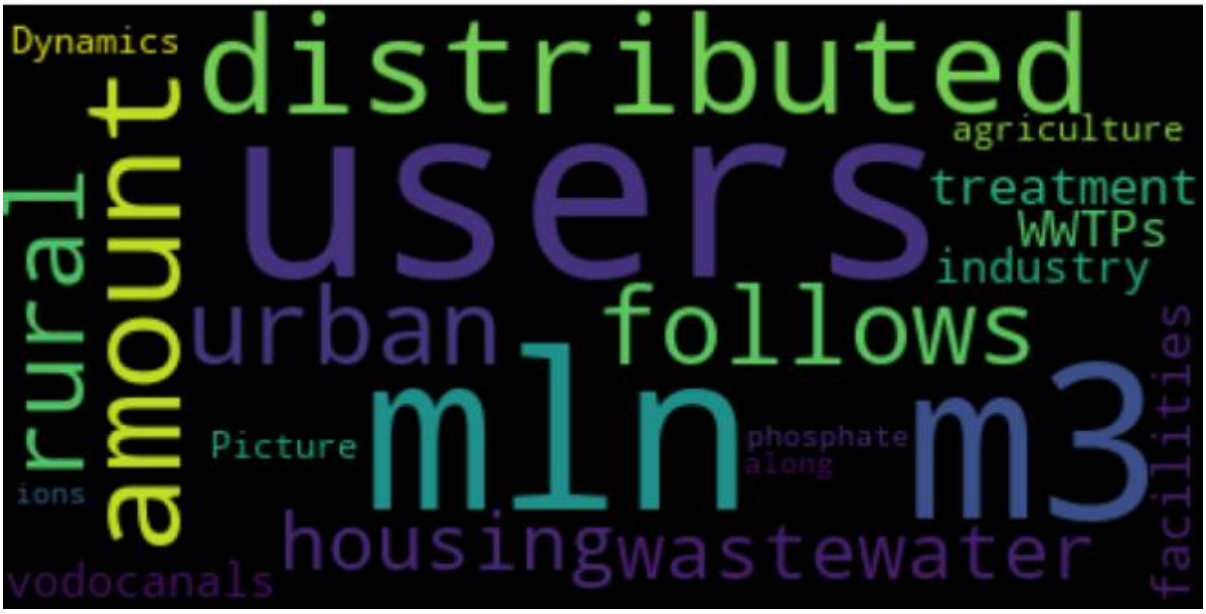

Рисунок Г.3 – Хмара слів по критерію «Pollution»

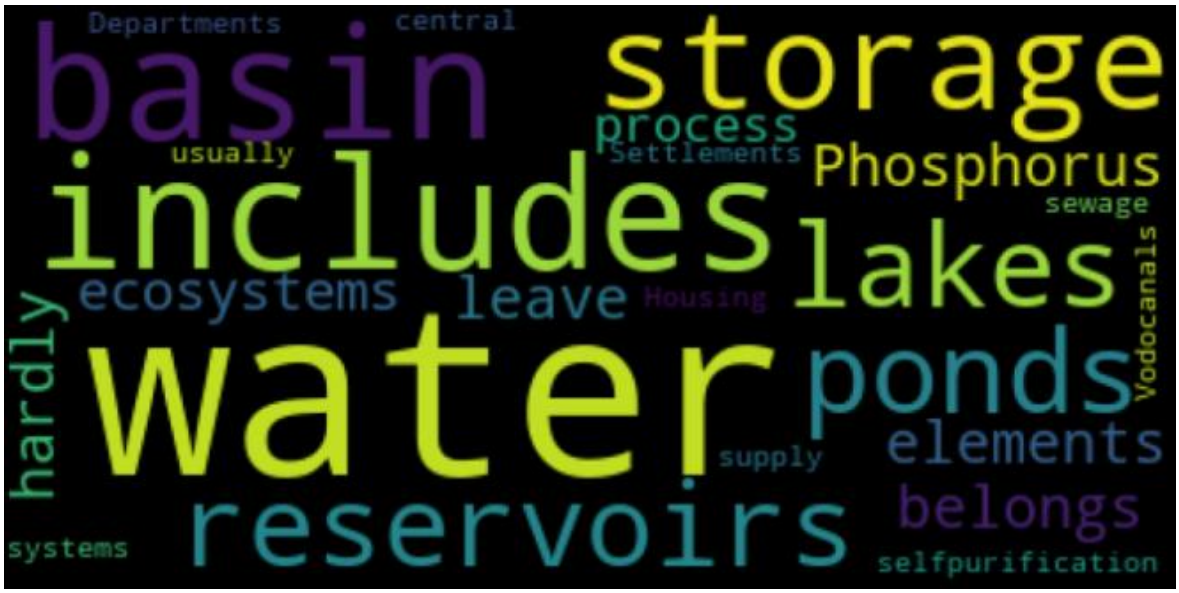

Рисунок Г.4 – Хмара слів по параметру «Not Pollution»

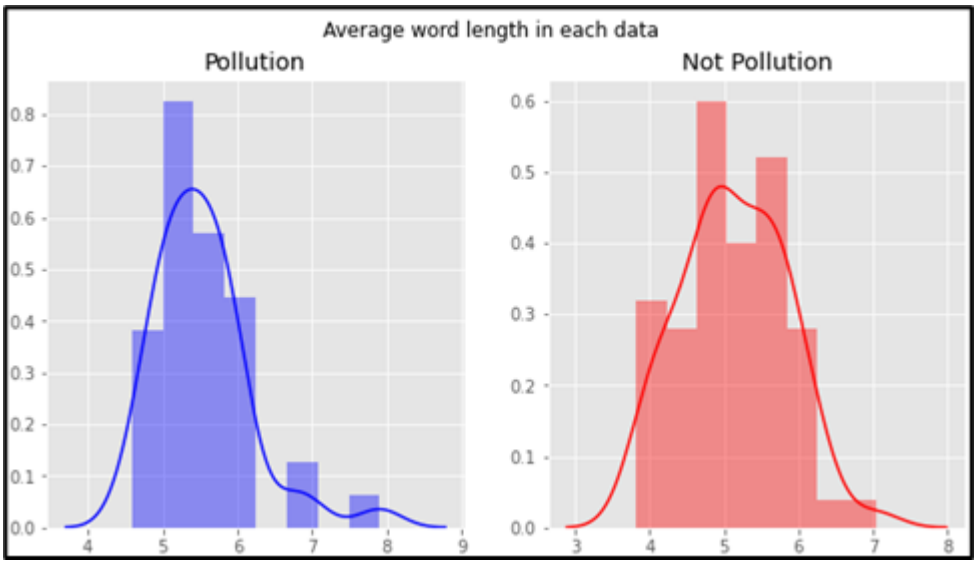

Рисунок Г.5 – Середня довжина слів

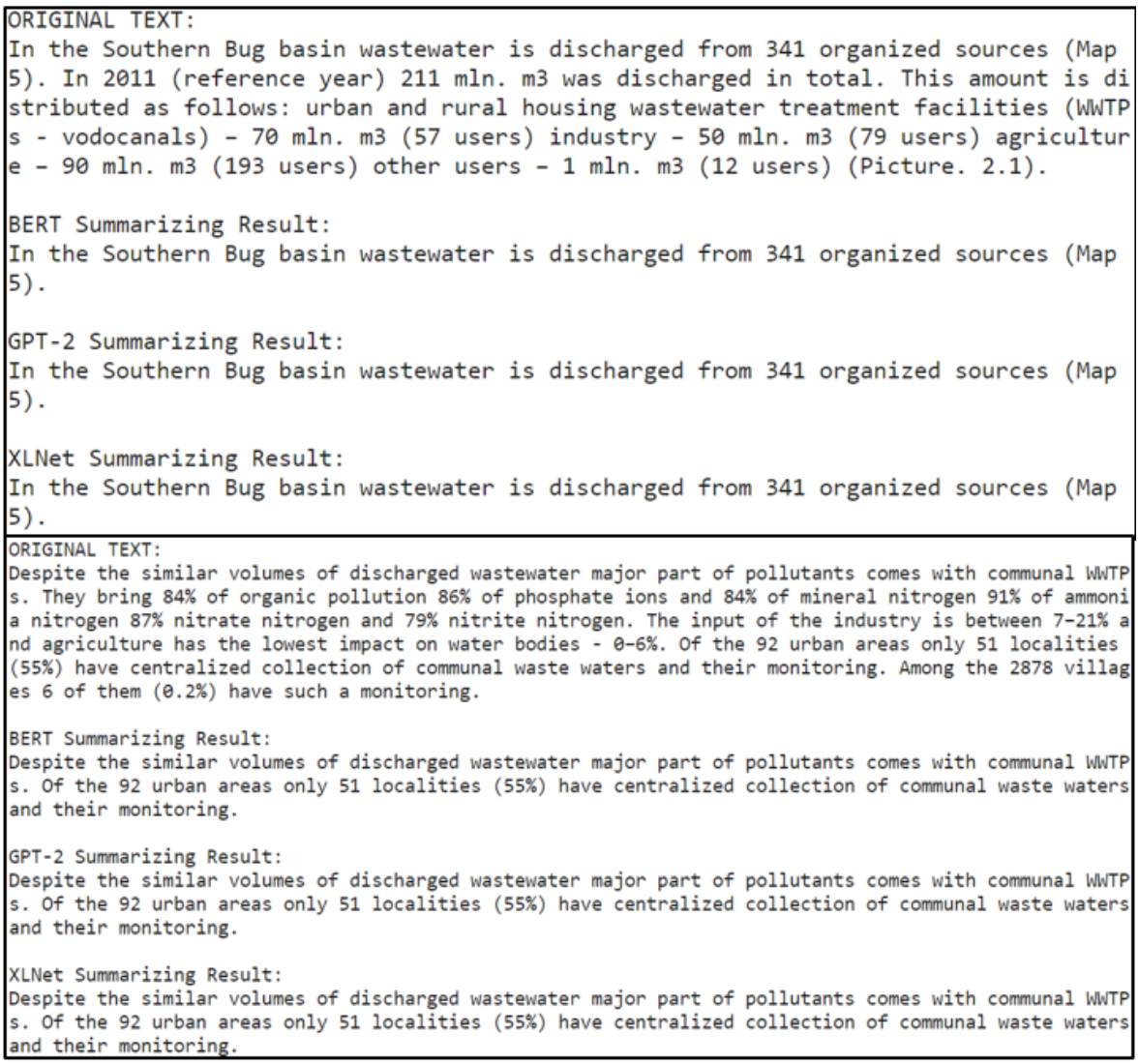

Рисунок Г.6 – Результати анотування тексту №1

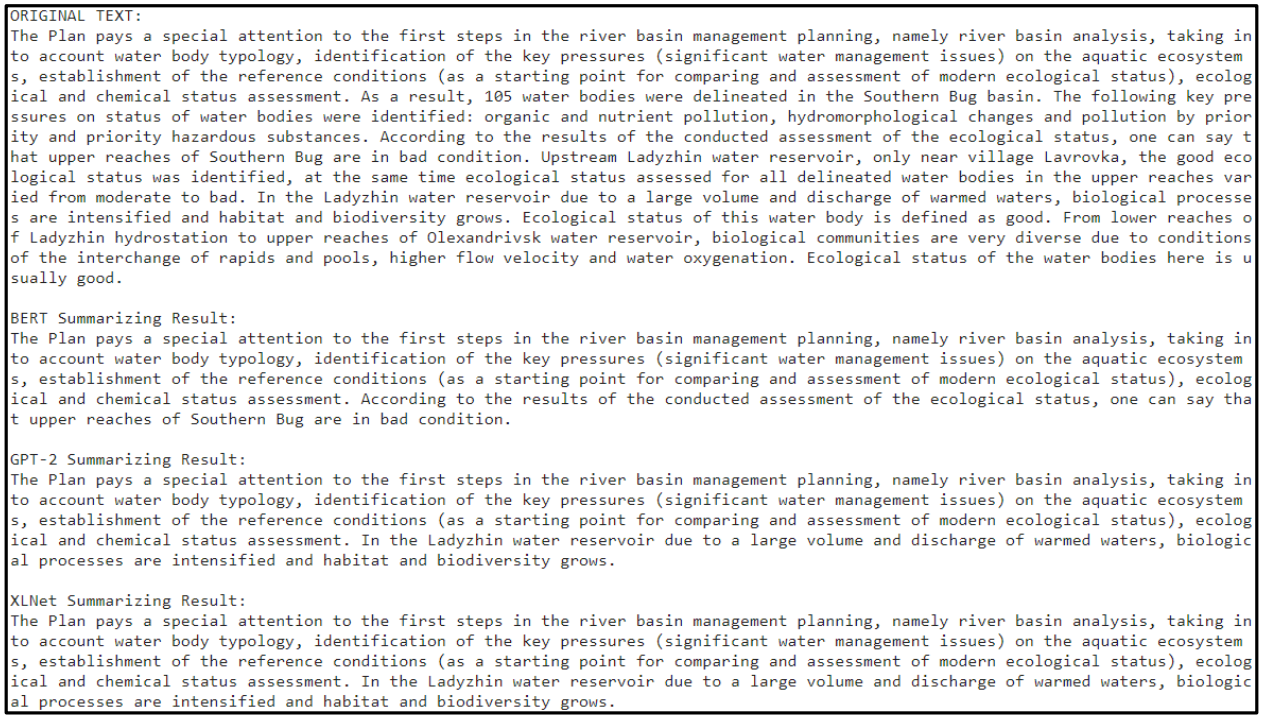

Рисунок Г.7 – Результати анотування тексту №2

ORIGINAL TEXT:
The degree of connectivity of the population to WWTPs varies widely - from 1 1% to 95%. The maximum rate is observed in Pivdennoukrainsk. In large cities with EH close or greater than 100000 EH connection to sewers ranges 85-92 8 0% in towns usually no more than 30% in rural settlements. Due to lack of da ta no accurate estimates can be given. Based on expert judgment taking that average degree of connectivity of the urban population to WWTPs is approxima tely 80%. The total load of inorganic compounds of nitrogen produced in 2011 at wastewater facilities with >2000 was more than 488 tons and their spatial distribution. The obtained results show that the main part (77%) of inorgani c nitrogen comes to waste waters from three the biggest oblast centers вЂ“ V innitsya Khmelnitsky and Kirogovrad. The majority of the wastewater treatmen t facilities in the Southern Bug basin are equipped with the facilities for biological treatment of wastewaters. This method allows extracting only a pa rt of pollutants containing nitro en and phosphorus. Technologies of deep ex traction of these compounds are not used in Ukraine due to their high cost. The existing wastewater treatment facilities were constructed during the Sov iet time they are physically and technologically outdated they work ineffici ently. Very often the volumes of wastewater exceed designed values which lea d to additional pollution. Most of wastewater treatment facilities require u rgent modernization and reconstruction. According to the obtained data waste water facilities with the lowest efficiency regarding Ninorg treatment are i n Vatutino Tulchin Bashtanka Smoline settlements. Wastewater facilities of t he large cities (> 100000) namely Kirovograd Vinnitsya Khmelnitsky also wor k ineffectively and lead to increased contains of nitrogen in the surface wa ters.

BERT Summarizing Result:
The degree of connectivity of the population to WWTPs varies widely - from 11% to 95%. Based on expe rt judgment taking that average degree of connectivity of the urban population to WWTPs is approxima tely 80%. Technologies of deep extraction of these compounds are not used in Ukraine due to their hi gh cost. According to the obtained data wastewater facilities with the lowest efficiency regarding N inorg treatment are in Vatutino Tulchin Bashtanka Smoline settlements.

GPT-2 Summarizing Result:
The degree of connectivity of the population to WWTPs varies widely - from 11% to 95%. Very often th e volumes of wastewater exceed designed values which lead to additional pollution. Wastewater facili ties of the large cities (> 100000) namely Kirovograd Vinnitsya Khmelnitsky also work ineffectively and lead to increased contains of nitrogen in the surface waters.

XLNet Summarizing Result:
The degree of connectivity of the population to WWTPs varies widely - from 11% to 95%. Based on expe rt judgment taking that average degree of connectivity of the urban population to WWTPs is approxima tely 80%. Very often the volumes of wastewater exceed designed values which lead to additional pollu tion. Wastewater facilities of the large cities (> 100000) namely Kirovograd Vinnitsya Khmelnitsky also work ineffectively and lead to increased contains of nitrogen in the surface waters.

Рисунок Г.8 – Результати анотування тексту №3