Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра прикладної математики
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Застосування нейронних мереж до розв'язання задачі тематичного моделювання
(тема)

Виконав:
студент _2_ курсу, групи _САУм-22-1
Стецун К.С.
(прізвище, ініціали)

Спеціальність
124 Системний аналіз
(код і повна назва спеціальності)
Тип програми __освітньо-професійна
Освітня програма
Системний аналіз і управління
(повна назва освітньої програми)

Керівник доц. Гибкіна Н.В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ПМ _____ Сидоров М.В.
(підпис) (прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет _інформаційно-аналітичних технологій та менеджменту
Кафедра _прикладної математики
Рівень вищої освіти _другий (магістерський)
Спеціальність _124 Системний аналіз
(код і повна назва)
Тип програми_освітньо-професійна
Освітня програма _Системний аналіз і управління
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри ПМ _____
(підпис)
"_06_" _листопада_ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові _Стецун Катерині Сергіївні
(прізвище, ім'я, по батькові)

1. Тема роботи _Застосування нейронних мереж до розв'язання задачі тематичного моделювання

затверджена наказом по університету від 2 листопада 2023 р. № 1277 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 10 січня 2024 р.

3. Вихідні дані до роботи _колекція текстів наукової спрямованості

4. Перелік питань, що потрібно опрацювати в роботі
1. Системний аналіз предметної області
2. Вибір і обґрунтування методу розв'язання
3. Програмна реалізація
4. Результати обчислювального експерименту
5. Аналіз можливих застосувань

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій

1. Актуальність теми роботи
2. Постановка задачі
3. Системний аналіз предметної області
4. Метод чисельного аналізу
5. Результати обчислювального експерименту

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Підбір та вивчення технічної літератури за темою роботи	6 – 12 листопада 2023 р.	виконано
2	Вибір та обґрунтування методу	13 – 26 листопада 2023 р.	виконано
3	Розробка алгоритму і програми	27 листопада – 10 грудня 2023 р.	виконано
4	Проведення аналітичних досліджень та розрахунків	11 грудня – 24 грудня 2023 р.	виконано
5	Робота над текстом пояснювальної записки	25 грудня 2023 р. – 9 січня 2024 р.	виконано
6	Представлення роботи на рецензію в ЕК	10 січня 2024 р.	виконано

Дата видачі завдання _6_ листопада 2023 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Гибкіна Н.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 80 с., 18 табл., 14 рис., 1 дод., 14 джерела.

ТЕМАТИЧНА МОДЕЛЬ, НЕЙРОННА МЕРЕЖА, МОДЕЛЬ BAT, МІШОК СЛІВ, ЕПОХА.

Об’єкт дослідження – задача тематичного моделювання.

Мета роботи – аналіз використання нейронних мереж у розв’язанні задачі тематичного моделювання наукових текстів.

Методи дослідження – методи попередньої обробки тексту, методи векторного подання документів («Bag-of-Words»), нейронна мережа BAT.

У кваліфікаційній роботі розглянуто задачу тематичного моделювання. У ході проведення системного аналізу для розв’язання задачі тематичного моделювання наукових текстів було обрано метод, що базується на застосуванні нейронної мережі. На основі обраного методу було розроблено програмний продукт для аналізу наукових текстових документів, проведено обчислювальні експерименти та проаналізовано отримані результати.

ABSTRACT

Introductory note: 80 pages, 18 tables, 14 figures, 1 appendix, 14 sources.

TOPIC MODEL, NEURAL NETWORKS, BAT MODEL, BAG OF WORDS, EPOCH.

Object of research – problem of thematic modeling.

Purpose of work – an analysis of the use of neural networks in solving the problem of thematic modeling of scientific texts.

Methods of research – methods of text processing, methods of vector presentation of documents ("Bag-of-Words"), BAT neural network.

The qualification work addresses the task of topic modeling. During the course of a systematic analysis for solving the problem of topic modeling in scientific texts, a method based on the application of a neural network was chosen. Based on the selected method, a software product for analyzing scientific textual documents was developed, computational experiments were conducted, and the obtained results were analyzed.

ЗМІСТ

ВСТУП

Актуальність теми. В наш час можливості обробки та аналізу інформації стикаються з суттєвим викликом через зріст обсягу даних, які генеруються різними джерелами, та зменшення часових ресурсів самої людини на їх аналіз. Навіть у галузі наукових публікацій, яка вже давно є невід'ємною складовою продукування нових знань, величезна кількість статей та досліджень може призвести до втрати орієнтації під час пошуку важливої інформації. Необхідна інформація може бути розсіяною або непомітною через великі обсяги даних, що стимулює необхідність вдосконалення методів її виявлення та ефективної обробки. У зв'язку з цим ефективне використання інноваційних технологій, зокрема нейромережевих, стає ключовим елементом для подолання такого інформаційного перенавантаження. Ці технології можуть висвітлити та систематизувати важливі аспекти інформації, що забезпечить вченим та дослідникам інструмент для більш ефективного аналізу і отримання нових знань в умовах сучасної інформаційної епохи. При дослідженні текстової інформації вони не лише здатні автоматизувати процеси обробки та аналізу даних, але і виявляти складні залежності та визначати ключові теми в потоці інформації.

Однією з ключових проблем при вирішенні цих задач є обробка та аналіз неструктурованих даних, таких як статті, наукові публікації та соціальні медіа-повідомлення. У цьому контексті нейронні мережі демонструють свою ефективність. Їхня здатність до навчання на великих обсягах даних та автоматичного виявлення складних залежностей робить їх потужним інструментом для виділення тематичних структур у тексті. Процеси автоматизації виділення тем та аналізу тексту, що реалізуються з використанням нейронних мереж, можуть суттєво прискорити та покращити наукові дослідження, забезпечуючи більш глибоке розуміння контенту. Тому попит на алгоритми та програми, що допомагають розв'язувати задачу аналізу текстів, також зростає, а самі програми стають все складнішими та мають все більше вимог для реалізації.

Мета і завдання кваліфікаційної роботи. Дана кваліфікаційна робота присвячена дослідженню застосування нейронних мереж до розв'язання задачі тематичного моделювання наукових текстів.

Для досягнення поставленої мети необхідно виконати наступні завдання:

– провести огляд і аналіз сучасного стану задачі ймовірнісного тематичного моделювання;

– розв’язати задачу тематичного моделювання за допомогою нейронної мережі;

– провести обчислювальні експерименти з вхідними даними, отриманими з різноманітних текстових джерел наукової інформації;

– провести аналіз отриманих результатів, зокрема порівняти якість розбиття документів за темами з фактичними темами, до яких належать аналізовані документи.

Об’єктом дослідження є задача тематичного моделювання.

Предметом дослідження є застосування нейронної мережі до розв'язання задачі тематичного моделювання наукових текстів.

Методи дослідження. У кваліфікаційній роботі використовуються методи попередньої обробки тексту, методи векторного подання документів («Bag-of-Words»), нейронна мережа BAT.

Публікації. Результати, отримані у роботі, було представлено на 26-му міжнародному молодіжному форумі «Радіоелектроніка і молодь у XXI столітті» (20 грудня 2022 р.) [11], Міжнародній науково-практичній конференції «Інформаційні технології та комп’ютерне моделювання» (15-16 грудня 2022 р.) [12], 27-му Міжнародному молодіжному форумі «Радіоелектроніка та молодь у XXI столітті» (м. Харків, 10-12 травня 2023 р.) [13] та Міжнародній конференції «LEARNING & TEACHING: after War and during Peace» (10 листопада 2023) [14].

1 СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧ ДОСЛІДЖЕННЯ

1.1 Системний аналіз задачі тематичного моделювання наукових текстів

1.1.1 Вербальна модель системи

Для здійснення аналізу задачі тематичного моделювання спочатку визначимося з наступними складовими аналізу.

Об’єкт аналізу – «Готовий набір текстових документів».

Предмет аналізу – «Визначення тематики документів».

Точка зору: виконавець.

Ціль: застосування тематичного моделювання для розв’язання задачі визначення тематики наукового документу..

Метою тематичного моделювання є розподіл документів відповідно до виявленої тематичної кластерної структури, виявлення тем документів і аналіз структурних складових документів та їх тем. Даний спосіб моделювання набору текстових документів визначає, до якої з тем буде відноситися кожний з документів.

Гнучкість методу тематичного моделювання розкриває безмежні можливості для експериментів з поєднанням різних підходів до обробки даних, сприяючи досягненню оптимальних результатів. На відміну від кластерного аналізу, де документи розподіляються на групи, тематичне моделювання дозволяє виокремити основні теми корпусу документів, орієнтуючись на їх внутрішні особливості та ключові слова. Такий підхід стає особливо корисним в аналізі великих обсягів даних, де важливо визначити не тільки структуру, але й сутність інформації. Завдяки цьому дослідники можуть не лише виявити патерни в текстах, але і здійснювати ефективний контроль над процесом аналізу для досягнення максимально точних та важливих висновків.

1.1.2 Морфологічний опис системи

Система – це комплекс взаємопов’язаних між собою елементів, котрі утворюють єдине ціле. В нашому випадку системою буде тематична модель.

Призначення системи – визначення тематики та ключових слів окремих текстових документів на практиці.

Мета системи – розподіл за темами текстових документів з заданого набору.

Проведемо класифікацію системи. Для цього класифікуємо систему за наступними пунктами [1]:

– за походженням: штучна (система зроблена людьми);

– за об’єктивністю існування: абстрактна (система є сукупністю математичних моделей і методів);

– за природою систем: механічна (відноситься до неживих систем);

– за центром системи: централізована (головну роль в системі відіграє дослідник, адже без нього система не зможе існувати);

– за величиною: мала;

– за складністю: складна;

– з точки зору взаємодії з оточуючим середовищем: відкрита (оскільки система обробляє наукові тексти, що беруться з зовнішнього середовища, тобто здійснюється обмін інформацією);

– за степенем детермінованості: детермінована (всі виходи можливо передбачити на початку роботи системи);

– за способом управління: самоврядна (керуючий блок буде знаходитися всередині системи, так як виконавець є частиною системи);

– за станом: статична система (системі не притаманні зміни з плином часу і вона залишається в одному й тому ж стані).

Дана система буде оптимальною, оскільки всі процеси в ній направлені на оптимізацію її роботи і витрат ресурсів (часу на написання алгоритму, часу роботи програми тощо) для отримання остаточного результату (розв’язання поставленої задачі).

Ідентифікованість системи буде полягати в тому, яким саме методом буде розв’язана задача тематичного моделювання, це й буде характерною ознакою, яка буде виділяти нашу систему з інших схожих систем.

Проаналізуємо межі системи. В нашому випадку межею системи є простір, в котрому виконавець за допомогою власного комп’ютеру буде досліджувати задачу тематичного моделювання та створювати алгоритм для її розв’язання.

Перелічимо основні об’єкти зовнішнього середовища нашої системи, побудувавши схему «Система – Зовнішнє середовище» (рисунок 1.1). Дана схема є сукупністю всіх об’єктів поза межами системи, при змінюванні властивостей яких система також буде змінюватися, а також об’єктів, властивості яких будуть змінюватися під впливом змінювання системи.

Рисунок 1.1 – Модель зовнішнього середовища

Далі схематично зобразимо модель «чорної скрині» (рисунок 1.2), що містить межі системи, модель зовнішнього середовища, а також найсуттєвіші входи й виходи системи.

Модель «чорної скрині» є системою, головна ідея якої полягає в тому, що дослідник може чітко визначити вхідні та вихідні дані, що будуть впливати на систему, проте внутрішні процеси, що відбуваються всередині скрині, залишаються невідомими [1, 2].

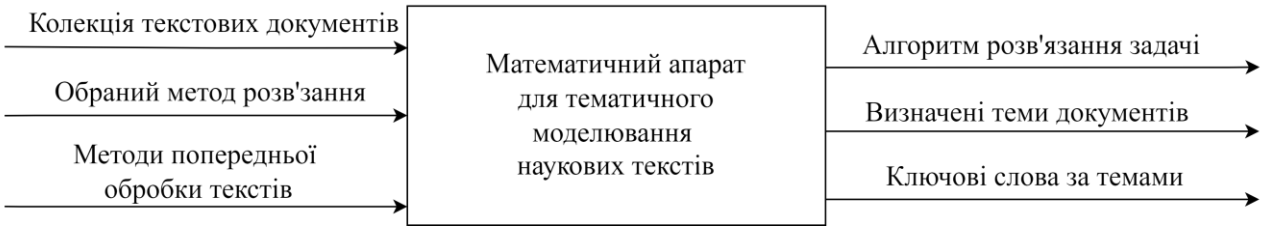


Рисунок 1.2 – Схема моделі «чорна скриня»

В нашій системі входами, тобто ресурсним впливом зовнішнього середовища, будуть: готовий набір документів, обраний спосіб моделювання та методу попередньої підготовки текстів. Виходами системи, тобто її цільовим продуктом, будуть: готовий алгоритм для розв’язання задачі, визначені теми документів та ключові слова за темами.

1.2 Аналіз сценаріїв вирішення задачі тематичного моделювання наукових текстів

1.2.1 Модель аналізу проблеми

Сформулюємо проблему вибору методу, за допомогою якого будемо розв’язувати задачу тематичного моделювання наукових текстів, та розв’яжемо її, використовуючи аналіз ієрархій [1, 3, 4]. Створимо модель інформаційного обміну організації, в нашому випадку, алгоритму розв’язання задачі тематичного моделювання, та зобразимо декомпозицію DFD-діаграми першого рівня на рисунку 1.3.

DFD-діаграма – це діаграма потокових даних, котра виступає в ролі одного з основних інструментів структурного системного аналізу та проектування інформаційних систем.

В такій системі процеси, що виконуватимуться в системі, будуть зображуватися прямокутниками з закругленими кутами. Такі блоки мають входи та виходи, але не підтримують управління та механізм. В кожен блок може входити та виходити по декілька стрілок.

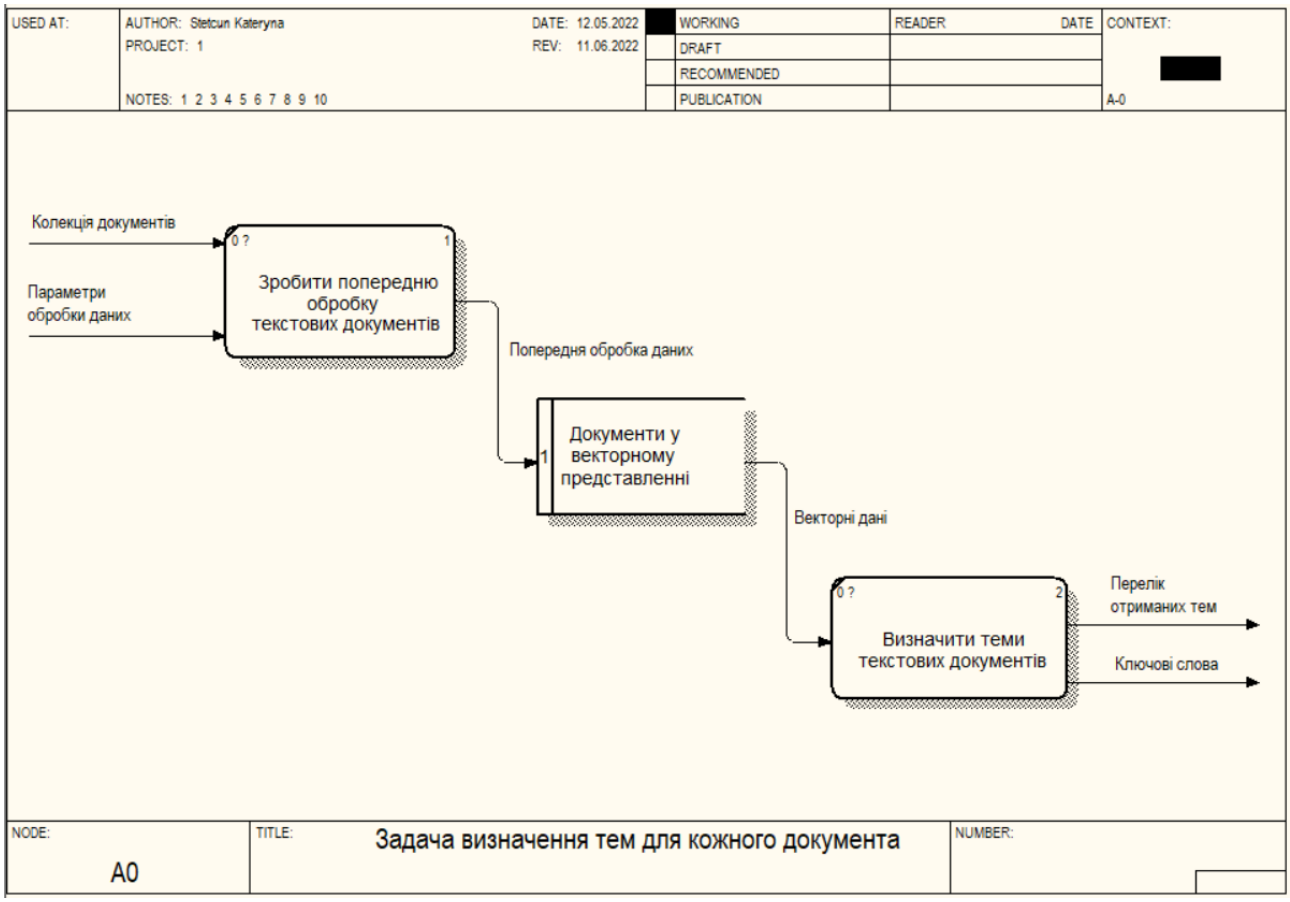


Рисунок 1.3 – Декомпозиція DFD-діаграми першого рівня

Далі проаналізуємо критерії, за якими будемо вирішувати поставлену проблему. У якості критеріїв вибору методу розв’язання задачі тематичного моделювання текстів оберемо наступні:

– критерій 1 (К1): комплексність використовуваного алгоритму;

– критерій 2 (К2): час роботи програми;

– критерій 3 (К3): точність отриманих результатів;

– критерій 4 (К4): обсяг пам’яті, що займає програма;

– критерій 5 (К5): універсальність обраного алгоритму.

Вибірка альтернатив для обрання методу розв’язання буде складатися з наступної множини значень:

– альтернатива 1 (А1): метод кластерного аналізу;

– альтернатива 2 (А2): метод ймовірнісного тематичного моделювання;

– альтернатива 3 (А3): нейронні мережі.

Ієрархічна структура для вирішення задачі тематичного моделювання наукових текстів має вигляд, поданий на рис. 1.4. Зауважимо, що в нашому випадку структура буде мати тип відображення – деревоподібний.

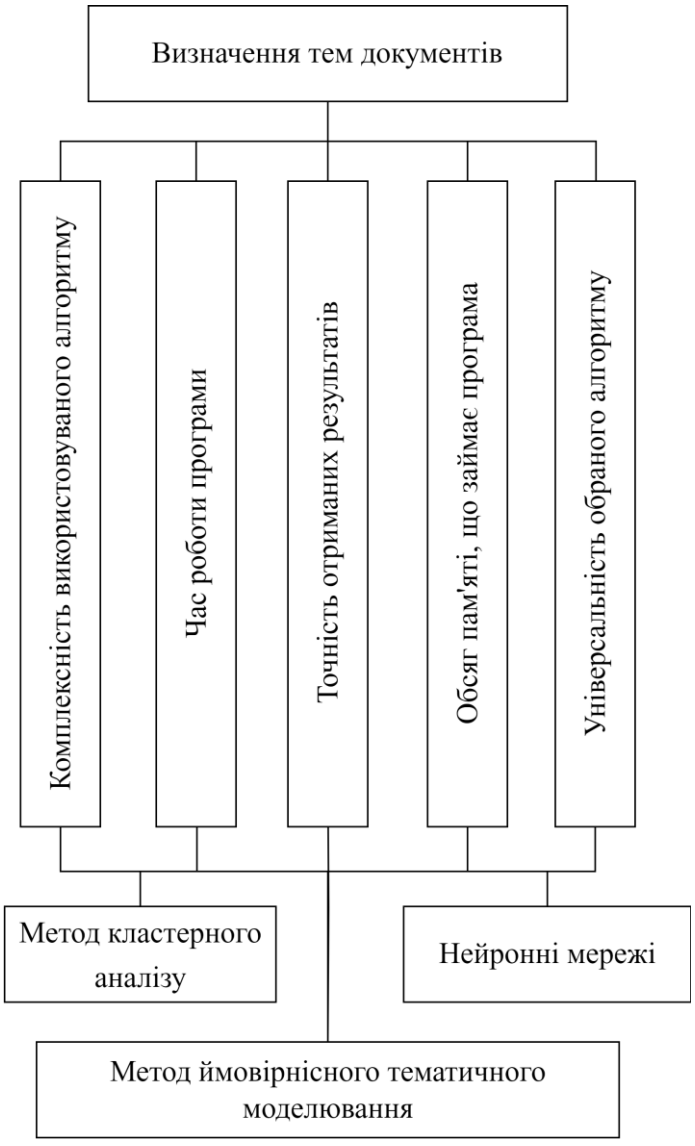


Рисунок 1.4 – Ієрархічна структура системи

1.2.2 Оцінювання вектора пріоритетів незадоволеностей методом аналізу ієрархій

Для аналізу сценаріїв вирішення задачі тематичного моделювання наукових текстів спершу побудуємо матрицю попарних порівнянь моделі, яка вказана в таблиці 1.1, а також матриці попарних порівнянь критеріїв системи.

Таблиця 1.1 – Матриця попарних порівнянь

Критерії оцінювання	К1	К2	К3	К4	К5	Оцінки компонентів	Вектор пріоритетів
К1	1	$\frac{1}{3}$	3	$\frac{1}{5}$	$\frac{1}{3}$	0,58	0,09
К2	3	1	1	$\frac{1}{6}$	3	1,08	0,17
К3	$\frac{1}{3}$	1	1	$\frac{1}{3}$	4	0,85	0,13
К4	5	6	3	1	5	3,39	0,53
К5	3	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{5}$	1	0,55	0,09
Всього						6,46	

Для таблиці 1.1 індекс узгодженості $(\text{ІУ}) = \frac{5,78-5}{5-1} = 0,196$, випадкова узгодженість дорівнює 1,12.

Тоді відносна узгодженість $(\text{ВУ}) = \frac{0,196}{1,12} = 0,175 = 17,5\%$.

Проведемо порівняльний аналіз альтернатив, щоб визначити, яка з них буде найкращою в даному випадку. Для цього кожну з альтернатив оцінимо, отримавши дані, наведені в таблицях 1.2 – 1.7. Зауважимо, що для наведених нижче таблиць випадкова узгодженість буде дорівнювати 0,58.

Таблиця 1.2 – Порівняння за першим критерієм

Критерій 1	A1	A2	A3	Власний вектор	Вектор пріоритетів
A1	1	5	4	2,71	0,63
A2	$\frac{1}{5}$	1	$\frac{1}{8}$	0,29	0,07
A3	$\frac{1}{4}$	8	1	1,26	0,30
Всього				4,15	

Для таблиці 1.2 $\text{ІУ} = \frac{3,28-3}{3-1} = 0,139$, а $\text{ВУ} = \frac{0,139}{0,58} = 0,241 = 24,1\%$.

Таблиця 1.3 – Порівняння за другим критерієм

Критерій 2	A1	A2	A3	Власний вектор	Вектор пріоритетів
A1	1	6	4	2,46	0,59
A2	$\frac{1}{6}$	1	3	0,29	0,07
A3	$\frac{1}{4}$	$\frac{1}{3}$	1	1,39	0,33
Всього				4,16	

Для таблиці 1.3 $\text{ІУ} = \frac{3,28-3}{3-1} = 0,139$, а $\text{ВУ} = \frac{0,139}{0,58} = 0,241 = 24,1\%$.

Таблиця 1.4 – Порівняння за третім критерієм

Критерій 3	А1	А2	А3	Власний вектор	Вектор пріоритетів
А1	1	$\frac{1}{7}$	$\frac{1}{5}$	0,31	0,07
А2	7	1	$\frac{1}{3}$	1,33	0,32
А3	5	3	1	2,47	0,60
Всього				3,74	

Для таблиці 1.4 $\text{ІУ} = \frac{3,029-3}{3-1} = 0,015$, а $\text{ВУ} = \frac{0,015}{0,58} = 0,025 = 2,5\%$.

Таблиця 1.5 – Порівняння за четвертим критерієм

Критерій 4	А1	А2	А3	Власний вектор	Вектор пріоритетів
А1	1	$\frac{1}{8}$	$\frac{1}{4}$	0,36	0,08
А2	8	1	5	0,84	0,19
А3	4	$\frac{1}{5}$	1	3,27	0,73
Всього				4,48	

Для таблиці 1.5 $\text{ІУ} = \frac{3,06-3}{3-1} = 0,032$, а $\text{ВУ} = \frac{0,032}{0,58} = 0,056 = 5,6\%$.

Таблиця 1.6 – Порівняння за п’ятим критерієм

Критерій 5	A1	A2	A3	Власний вектор	Вектор пріоритетів
A1	1	4	$\frac{1}{7}$	0,83	0,19
A2	$\frac{1}{4}$	1	$\frac{1}{5}$	0,37	0,08
A3	7	5	1	3,27	0,73
Всього				4,47	

Для таблиці 1.6 $ІУ = \frac{3,34-3}{3-1} = 0,169$, а $ВУ = \frac{0,169}{0,58} = 0,292 = 29,2\%$.

1.2.3 Модель вирішення проблеми

Проаналізувавши отримані значення, особа, що приймає рішення, тобто ми в даному випадку, може зробити висновки щодо найкращої альтернативи для вирішення поставленої перед нею задачі.

З використанням вектору пріоритетів критеріїв, обчисленого у таблиці 1.1, та векторів пріоритетів за окремими критеріями (таблиці 1.2 – 1.6) виконаємо розрахунок вектору глобальних пріоритетів. Результати розрахунків наведені у таблиці 1.7.

Аналізуючи розраховані значення вектору глобальних пріоритетів, можемо зробити висновок, що в умовах заданого набору критеріїв доцільно буде серед наявних альтернатив вибрати третю, а саме метод нейронних мереж.

Таблиця 1.7 – Глобальні пріоритети

Критерій /Альтернатива	К1	К2	К3	К4	К5	Глобальні пріоритети
А1	0,636	0,363	0,075	0,068	0,186	0,18
А2	0,069	0,843	0,324	0,733	0,083	0,58
А3	0,295	3,271	0,602	0,199	0,732	0,82

1.3 Змістовна та формальна постановка задачі

1.3.1 Змістовна постановка задачі

На основі інформації, зібраної під час аналізу вирішення поставленої проблеми, сформулюємо змістовну постановку задачі. У нашому випадку вона полягає у визначенні тематики окремих документів із сформованої колекції документів. Документи колекції є текстовими документами наукової спрямованості, зібраними з відкритих джерел.

Метою тематичного моделювання є побудова моделі набору документів, яка дозволить визначати теми для кожного з документів та ключові слова за визначеними темами. Зауважимо, що хоч, на перший погляд, тематичне моделювання й схоже на кластеризацію, проте одна з головних відмінностей полягає в тому, що при тематичному моделюванні документи можуть буди віднесені одразу до декількох класів, тобто належати до декількох тем, а не відноситися чітко до однієї теми. Отже, остаточним результатом тематичного моделювання буде набір документів, кожен з яких буде помічений декількома темами з загального списку. Кожній виділеній темі відповідатиме набір ключових слів, тобто таких слів, що найбільш характерні для цієї теми.

Оскільки набір даних для аналізу являє собою звичайні текстові документи, написані природною мовою, перед проведенням подальшого аналізу важливо виконати попередню обробку цих документів. Ця процедура

націлена на оптимізацію роботи моделі, спрощення реалізації алгоритму та підвищення продуктивності програмного забезпечення. До простих, але важливих методів попередньої обробки, відносяться лематизація, стемінг, вилучення рідкісних слів, стоп-слів, цифр, гіперпосилань, слів з помилками, специфічних символів, зображень для досягнення кращих результатів у подальшому аналізі та моделюванні.

1.3.2 Формальна постановка задачі

Для розв'язання задачі тематичного моделювання використовуватимемо ймовірнісну тематичну модель.

Нехай D – колекція текстових документів, а d – окремі документи цієї колекції. Позначимо через W множину унікальних термів (слів, словосполучень, термінів) колекції D, інакше кажучи, словник. В такому випадку w – окреме слово (терм) в словнику. Множину тем позначимо через T, а символом t позначатимемо окремі теми з цієї множини. Терми w та документи d є спостережуваними змінними, а теми t – прихованими змінними [6].

Вважається, що поява термів w у документі d залежить від його теми t, але не від документа d, тобто:

$$p(w|d,t) = p(w|t).$$

Ймовірнісна тематична модель описує кожну тему дискретним розподілом ймовірностей слів $p(w|t)$, а кожен документ – дискретним розподілом ймовірностей тем $p(t|d)$ і передбачає, що розподіл термів у документі $p(w|d)$ визначається розподілами термів за темами $p(w|t)$ та тем за документами $p(t|d)$:

$$p(w|d) = \sum_{t\in T} p(w|t,d)\,p(t|d) = \sum_{t\in T} p(w|t)\,p(t|d). \qquad (1.1)$$

Задача тематичного моделювання полягає в тому, щоб знайти такі розподіл термів за темами $p(w|t)$ та розподіл множини тем $p(t|d)$, за яких тематична модель (1.1) якнайкраще наближає частотні оцінки ймовірностей $\hat{p}(w|d)$, обчислені за заданою колекцією документів D.

Таким чином, тематичне моделювання дозволяє визначати тематики документів колекції у вигляді розподілу $p(t|d)$, а також структуру кожної теми у вигляді розподілу слів $p(w|t)$.

Для можливості застосування тематичного моделювання необхідно подавати текстові дані у цифровому вигляді. Отже, на початковому етапі досліджувану колекцію документів D необхідно перетворити до векторного вигляду, який би відбивав особливості кожного окремого документа [7].

Для цього будемо використовувати модель Bag of Words, згідно з якою кожен документ подається у вигляді вектора $x \in \mathbb{N}^V$ з заданим розподілом ймовірностей, де V – розмір словника.

Результатом розв'язання задачі тематичного моделювання будуть ймовірності належності кожного документа до визначених алгоритмом тем та ключові слова за цими темами.

1.4 Постановка задач дослідження

Метою даної кваліфікаційної роботи є дослідження застосування нейронних мереж до розв'язання задачі тематичного моделювання наукових текстів.

Проаналізувавши всі отримані в попередніх трьох підпунктах дані, сформуємо перелік задач, які необхідно виконати під час дослідження:

– сформулювати задачу тематичного моделювання;

– розв'язати задачу тематичного моделювання за допомогою нейронних мереж;

– провести обчислювальні експерименти з вхідними даними, отриманими з різноманітних текстових джерел, з використанням різних методів попередньої обробки текстів;

– провести аналіз отриманих результатів, зокрема порівняти якість розбиття документів за темами з фактичними темами, до яких належать аналізовані документи.

2 ВИБІР ТА ОБҐРУНТУВАННЯ МЕТОДУ РОЗВ'ЯЗАННЯ

2.1 Основні відомості з тематичного моделювання

2.1.1 Постановка задачі тематичного моделювання

Тематичне моделювання використовується як інструмент для організації, аналізу, систематизації та впорядкування великих наборів текстової інформації. Воно представляє собою спосіб побудови моделі колекції текстових документів, яка дозволяє за сукупністю документів або сукупністю слів у документах отримати набір тем, що характеризують зміст досліджуваних документів. Для цього тематична модель має містити приховані змінні (що й керують семантикою досліджуваного документа), а мета моделювання полягає в тому, щоб розкрити ці приховані змінні, тобто теми, які визначають зміст документу. Дослідження в даному випадку відбувається шляхом спостереження за тим, які слова з'являються поруч з іншими словами, в яких темах, та збору цієї інформації з використанням статистики ймовірностей [5].

Розглянемо колекцію текстових документів D, яка складається з документів d. Нехай W – множина унікальних термів (слів, словосполучень, термінів) цієї колекції D, тобто словник, а w – окремі слова (терми) в словнику. Множина тем колекції позначається T і кожен її елемент t – це окрема тема, до якої можуть належати документи колекції. Теми t вважаємо прихованими змінними, а терми w та документи d – змінними, що спостерігаються [6].

Ймовірнісна тематична модель описує кожну тему дискретним розподілом ймовірностей слів $p(w|t)$, а кожен документ – дискретним розподілом ймовірностей тем $p(t|d)$. Вважається, що поява термів w у документі d залежить від його теми t, і не залежить від документа d, тобто:

$$p(w|d,t) = p(w|t).$$

Розподіл термів у документі $p(w|d)$ визначається розподілами термів за темами $p(w|t)$ та тем за документами $p(t|d)$:

$$p(w|d) = \sum_{t \in T} p(w|t,d)\,p(t|d) = \sum_{t \in T} p(w|t)\,p(t|d). \qquad (2.1)$$

За таких умов задача тематичного моделювання полягає в тому, щоб знайти такі розподіл термів за темами $p(w|t)$ та розподіл множини тем $p(t|d)$, за яких тематична модель (2.1) якнайкраще наближатиме частотні оцінки ймовірностей $\hat{p}(w|d)$, обчислені за заданою колекцією документів D.

Визначені ймовірнісні розподіли $p(t|d)$ та $p(w|t)$ характеризують тематики документів колекції ($p(t|d)$) і структуру кожної теми у вигляді розподілу слів ($p(w|t)$) відповідно.

Для можливості застосування тематичного моделювання, як і інших математичних методів статистичної обробки і аналізу текстів, окремі документи колекції мають бути подані у цифровому вигляді. Тому досліджувану колекцію документів D необхідно попередньо перетворити до векторного вигляду, який би відбивав статистичні та тематичні особливості кожного окремого документа [7].

2.1.2 Основні припущення моделі «Bag-of-Words»

Одним з найвідоміших та простих способів подання текстових даних у вигляді вектору чисел є модель «Bag-of-Words» («Мішок слів»). Ця модель дозволяє виявляти ознаки та особливості тексту для використання їх під час обробки цього тексту, зокрема алгоритмами машинного навчання, та описує наявність слова в документі.

Модель «Мішок слів» включає в себе: словник відомих слів (список унікальних слів, присутніх у тексті) та міру присутності цих слів у тексті.

Головні особливості «Мішка слів» полягають у наступному:

– модель вказує лише на міру входжень у документ тих чи інших слів зі словника, але не враховує їх розташування в самому документі;

– інформація щодо порядку та структури слів в документі не береться до уваги;

– складність моделі залежить від кількості відомих слів та алгоритму обчислення міри їх наявності у документі [6].

Дана модель проста для розуміння та імплементації, що й зумовлює її популярність серед інших способів цифрового подання текстових даних. Але, незважаючи на свою простоту, вона може бути використана для виявлення відмінностей у схожих документах та розподілу за темами документів зі схожим набором слів.

Один з підходів полягає у тому, що у поданні «Мішок слів» кожен документ є вектором $x \in \mathbb{N}^V$, що має поліноміальний закон розподілу:

$$x \sim \text{Polinomial}(n, p),$$

де V – кількість унікальних слів, тобто розмір словника;

$n \in \mathbb{N}$ – параметр, який вказує, скільки спроб (слів) потрібно для створення документа;

$p \in [0,1]^V$ – вектор ймовірностей для кожного слова у сформованому словнику.

Також вважатимемо, що існують K ($K \in \mathbb{N}$) поліноміальних розподілів, з яких можуть бути створені документи. Тоді повний розподіл всієї колекції документів може бути поданий сумішшю цих K розподілів

$$p(x) = \sum_{k=1}^{K} \pi_k p_k(x),$$

де p_k – розподіл ймовірностей k-го поліноміального закону розподілу;

π_k – вага k-го розподілу у суміші, така що $\pi_k \geq 0$ для всіх k та $\sum_{i=1}^{K} \pi_k = 1$.

Якщо $X = \{x^{(i)}\}_{i=1}^{M}$ – множина документів, то документ з k-го розподілу $x^{(i)} \sim p(x^{(i)} \mid t_k = 1)$ – це вектор у кодуванні Bag of Words за умови прихованої змінної t:

$$p\left(x^{(i)} \mid t_k = 1\right) = \text{Polinomial}\left(n_i, \beta_k\right),$$

де $n_i \in \mathbb{N}$ – кількість слів у i-му документі, $i \in \{1, \ldots, M\}$;

$\beta_k \in [0,1]^V$, $k \in \{1, ..., K\}$, – вектор ймовірностей k-го поліноміального розподілу;

t_k – елементи вектора-індикатора компонент суміші, такі що $t \in \{0,1\}^K \sim \text{Polinomial}(1, \pi)$ з рядом розподілу $p(t) = \prod_{k=1}^{K} \pi_k^{t_k}$;

$\pi \in [0,1]^K$ – вектор параметрів.

Отже, маємо спостережувані випадкові величини x, приховані випадкові величини t та параметри $\pi \in [0,1]^K$ і $B = \{\beta_k\}_{k=1}^{K}$.

Закон розподілу моделі суміші може бути визначений наступним чином:

$$p(x; \pi, B) = \sum_{k=1}^{K} p(x, t_k = 1; \pi, B) = \sum_{k=1}^{K} p(x \mid t_k = 1; B)\, p(t_k = 1; \pi) =$$

$$= \sum_{k=1}^{K} \left(\frac{\Gamma\left(\sum_{i=1}^{V} x_i + 1\right)}{\prod_{i=1}^{V} \Gamma(x_i + 1)} \prod_{j=1}^{V} \beta_{jk}^{x_j} \right) \pi_k^{t_k}.$$

За такої постановки задача тематичного моделювання полягатиме у тому, щоб визначити такі значення параметрів π, B, що найточніше відповідають досліджуваній колекції документів з точки зору присутніх у ній тематик [8].

Зауважимо, що перед побудовою векторної моделі окремих документів тексти необхідно піддати попередній обробці, щоб забезпечити можливість їх подальшого опрацювання та аналізу.

2.1.3 Попередня обробка даних

Для коректної обробки текстових даних їх необхідно підготувати, очистити від несуттєвої інформації, зменшити обсяг інформації, яка буде використовуватися для подальшого аналізу [7]. Основні етапи підготовки тексту можна поділити на:

– розбиття тексту на слова;

– видалення пунктуаційних знаків;

– видалення електронних адрес, гіперпосилань, чисел, дат тощо;

– переведення усіх слів в нижній регістр;

– застосування фільтрації, видалення стоп-слів або слів, що майже не використовуються;

– застосування стемінгу або лематизації.

Розглянемо більш детально окремі етапи попередньої обробки текстів.

Видалення слів, що рідко зустрічаються в тексті, електронних адрес, гіперпосилань, чисел, дат, зведення слів до нижнього регістру, тощо націлене на скорочення розмірів словника, що в свою чергу зменшує кількість пам’яті, необхідної для обробки даних, та пришвидшує навчання алгоритму та роботу готового програмного продукту.

Стоп-слова – це слова, які не вносять вклад у більш глибоке розуміння фрази, тому вони не є корисними для тематичних моделей. До стоп-слів відносяться частки, суфікси, прийменники, сполучники, дієприкметники,

вставні слова, займенники, сполучники, комбінації букв тощо. Кількість таких слів зазвичай невелика у межах одного документа, тому їх видалення майже не впливає на розміри словника, проте помітно скорочує його довжину.

Стемінг – це процес обрізання слова шляхом відкидання допоміжних частин (суфікса, префікса або закінчення) з метою скорочення цього слова до його основи. Даний спосіб нагадує процедуру знаходження кореня слова, проте результат стемінгу часто відрізняється від морфологічного кореня слова.

Лематизація – це процес зведення кожного слова в тексті до його базової (словарної) форми (іменників – до називного відмінка однини; прикметників – до називного відмінка однини у чоловічому роді; дієслів, прислівників, дієприслівників – до дієслова в інфінітиві (невизначеній формі) недоконаного виду). Лематизація також дозволяє скорочувати простір використаних слів.

Зауважимо, що для різних природних мов більш ефективними є різні методики попередньої обробки, так, для української мови більш підходить лематизація, а для англійської чи французької – стемінг [6].

2.2 Нейронні тематичні моделі

2.2.1 Застосування нейронних мереж для розв'язання задач тематичного моделювання

Тематичне моделювання, основоположний компонент у різноманітних сферах, таких як обробка природної мови та системи рекомендацій вмісту, стикається з властивими складнощами у з’ясуванні прихованих моделей у великих наборах даних. Традиційні методології, незважаючи на свою досконалість, борються зі складними залежностями та контекстно-специфічними нюансами, повсюдно присутніми в наборах даних реального світу.

Нейронні мережі демонструють доволі гарні результату у задачах розпізнаванні складних моделей у великих наборах даних. Ця унікальна

здатність походить від їхньої архітектури, яка розпізнає залежності та контекстуальні нюанси. У сфері тематичного моделювання, де тонкощі мають першочергове значення, нейронні мережі виділяються як перспективні інструменти для розкриття прихованих структур.

Застосування принципів глибокого навчання надає нейронним мережам можливість швидкого та комплексного аналізу великих наборів даних. Цей атрибут є особливо корисним у тематичному моделюванні, де дослідження багатогранних тем вимагає глибокого розуміння, чого традиційні методи часто важко досягти. Також нейронні мережі самостійно отримують ієрархічні представлення даних, що є ключовою перевагою тематичного моделювання. Ця властива здатність сприяє тонкому розумінню тем, дозволяючи нейронним мережам розкривати складні зв’язки в тематичному вмісті. Ієрархічний підхід покращує розкриття моделі та забезпечує більш детальний тематичний аналіз.

Також нейронні мережи можна назвати універсальними завдяки їх плавній інтеграції різноманітних методів представлення даних, включаючи текст, зображення та послідовні дані. Ця можливість адаптації є революційною в тематичному моделюванні, де набори даних часто демонструють різноманітні модальності. Здатність синтезувати інформацію з різноманітних джерел позиціонує нейронні мережі як потужні інструменти для цілісного тематичного аналізу. Зазначимо також, що вони є адаптивними, завдяки чому вони не обмеженні в роботі лише з однорідними наборами даних. Їх здатність до безперервного навчання забезпечує розвиток розуміння тематичного змісту. Оскільки нейронні мережі навчаються на досвіді, вони поступово покращують свою здатність розпізнавати приховані структури, підвищуючи якість тематичного аналізу з часом.

Підсумовуючи, об’єднання нейронних мереж у сферу тематичного моделювання не лише означає зміну парадигми, але також означає трансформаційну траєкторію до глибшого розуміння складних наборів даних. Адаптивність, навчання та розпізнавання шаблонів нейронних мереж позиціонують їх як науковий рубіж у вічній гонитві за комплексними

тематичними представленнями, пропонуючи безпрецедентні шляхи для дослідження та відкриття в еволюційному ландшафті штучного інтелекту.

2.2.2. Модель BAT

Як показано на рисунку 2.4, модель BAT (Bidirectional Adversarial Topic model) складається з трьох компонентів [9]:

1) енкодер (E) приймає на вхід V-вимірне представлення документа $\vec{d}_r$, взяте з текстового корпусу C та перетворює його у відповідний K-вимірний розподіл тем $\vec{\theta}_r$;

2) генератор (G) приймає на вхід випадковий тематичний розподіл $\vec{\theta}_f$, взятий з апріорного розподілу Діріхле, і генерує V-вимірний розподіл fake-слів $\vec{d}_f$;

3) дискримінатор (D) приймає дійсну пару розподілу $\vec{p}_r = \left[\vec{\theta}_r; \vec{d}_r\right]$ і фальшиву пару розподілу $\vec{p}_f = \left[\vec{\theta}_f; \vec{d}_f\right]$ на вхід та відокремлює справжні пари розподілу від фальшивих; вихідні сигнали дискримінатора використовуються як сигнали контролю для вивчення E, G і D під час змагального навчання. Далі детальніше опишемо кожен компонент.

Мережа Енкодера. Енкодер вивчає функцію відображення для перетворення розподілу слів документа у розподіл тем документа. Як показано на лівій верхній панелі рисунка 2.4, він містить V-вимірний шар розподілу слів документа, S-вимірний шар представлення та K-вимірний шар розподілу тем документа, де V і K позначають розмір словника та номер теми відповідно.

Тобто для кожного документа d у текстовому корпусі E приймає на вхід представлення документа $\vec{d}_r$, зважене TF-IDF, що обчислюється за формулами:

$$\mathrm{tf}_{i,d} = \frac{n_{i,d}}{\sum_{\upsilon} n_{\upsilon,d}},$$

$$\text{idf}_i = \log \frac{|C|}{|C_i|},$$

$$\text{tf-idf}_{i,d} = \text{tf}_{i,d} \cdot \text{idf}_i,$$

$$d_r^i = \frac{\text{tf-idf}_{i,d}}{\sum_\upsilon \text{tf-idf}_{\upsilon,d}},$$

де $n_{i,d}$ – кількість входжень i-го слова в документі d;

$|C|$ – кількість документів у корпусі;

$|C_i|$ – кількість документів, які містять i-е слово в корпусі.

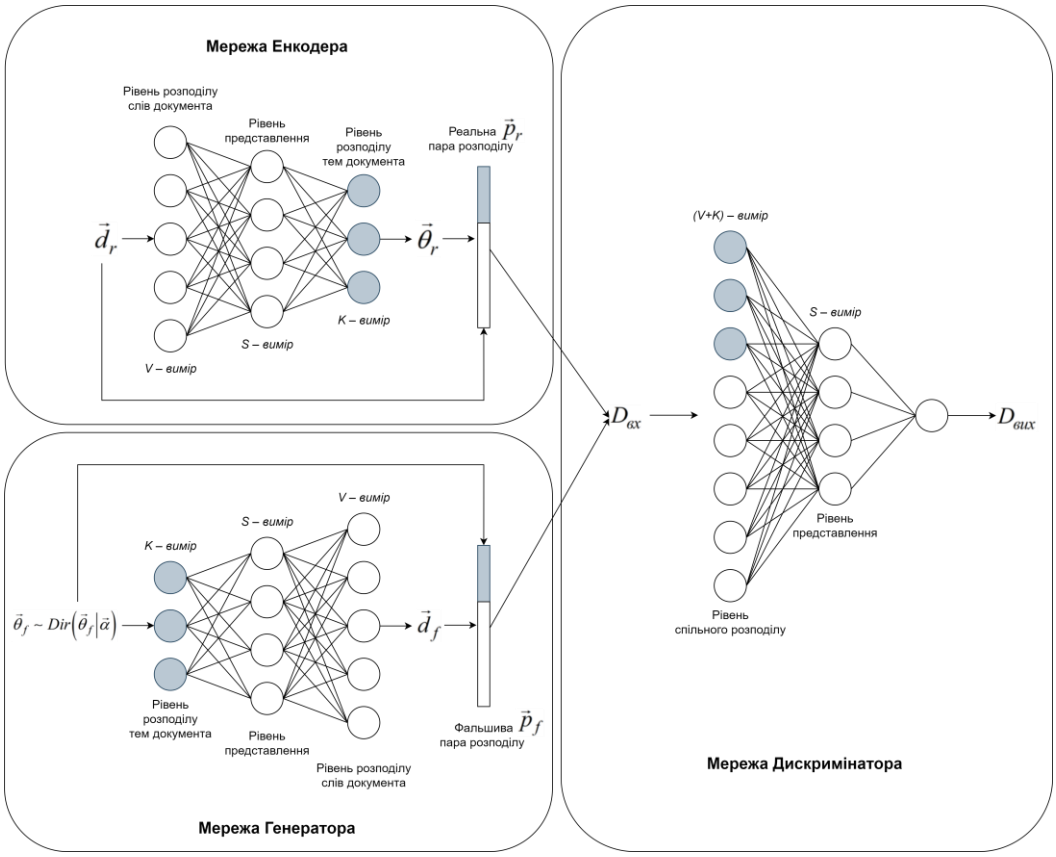


Рисунок 2.4 – Структура BAT

Таким чином, кожен документ можна представити V-вимірним поліноміальним розподілом, де i-а компонента позначає семантичну узгодженість між i-м словом і документом.

Використовуючи $\vec{d}_r$ як вхідні дані, E спочатку проектує його в S-вимірний семантичний простір через шар представлення:

$$\vec{h}_s^e = BN\left(W_s^e \vec{d}_r + \vec{b}_s^e\right),$$

$$\vec{o}_s^e = \max\left(\vec{h}_s^e, leak \cdot \vec{h}_s^e\right),$$

де $W_s^e \in \mathbb{R}^{S \times V}$ та $\vec{b}_s^e$ – вагова матриця та зміщення шару представлення;

$\vec{h}_s^e$ – вектор стану, нормалізований пакетною нормалізацією BN(·);

$leak$ – параметр активації LeakyReLU (Leaky Rectified Linear Unit);

$\vec{o}_s^e$ – результат шару представлення.

Енкодер перетворює $\vec{o}_s^e$ у K-вимірний тематичний простір на основі рівняння:

$$\vec{\theta}_r = \text{softmax}\left(W_t^e \vec{o}_s^e + \vec{b}_t^e\right),$$

де $W_t^e \in \mathbb{R}^{K \times S}$ – вагова матриця шару розподілу тем;

$\vec{b}_t^e$ – член зміщення;

$\vec{\theta}_r$ – відповідний розподіл тем вхідних даних $\vec{d}_r$, де k-й вимір θ_r^k, $k \in \{1, 2, ..., K\}$, представляє частку k-ї теми в документі d.

Мережа Генератора. Генератор показано на нижній лівій панелі рисунка 2.4. На відміну від енкодера, він забезпечує зворотну проекцію від розподілу тем документа до розподілу слів документа та містить K-вимірний шар теми документа, S-вимірний шар представлення та V-вимірний шар розподілу слів документа.

ВАТ використовує попередні параметри Діріхле, параметризовані $\vec{\alpha}$, щоб імітувати багатовимірний симплекс за тематичним розподілом $\vec{\theta}_f$. Його можна подати на основі рівняння:

$$p\left(\vec{\theta}_f \middle| \vec{\alpha}\right) = Dir\left(\vec{\theta}_f \middle| \vec{\alpha}\right) \triangleq \frac{1}{\Delta\left(\vec{\alpha}\right)} \prod_{k=1}^{K} \left[\theta_f^k\right]^{\alpha_k - 1},$$

де $\vec{\alpha}$ є K – вимірний гіперпараметр апріорного розподілу Діріхле;

K – номер теми, який слід встановити в BAT;

$\theta_f^k \in [0,1]$ – представляє частку k-ї теми в документі, $\sum_{k=1}^{K} \theta_f^k = 1.$

Член нормалізації $\Delta\left(\vec{\alpha}\right)$ визначається як $\frac{\prod_{k=1}^{K} \Gamma\left(\alpha_k\right)}{\Gamma\left(\sum_{k=1}^{K} \alpha_k\right)}$.

Щоб навчитися переходу від розподілу документів-тем до розподілу документів-слів, G спочатку проектує $\vec{\theta}_f$ у S-вимірний простір представлення на основі рівнянь:

$$\vec{h}_s^g = BN\left(W_s^g \vec{\theta}_f + \vec{b}_s^g\right),$$
$$\vec{o}_s^g = \max\left(\vec{h}_s^g, leak \cdot \vec{h}_s^g\right), \qquad (2.18)$$

де $W_s^g \in \mathbb{R}^{S \times K}$ – вагова матриця рівня представлення;

$\vec{b}_s^g$ – член зміщення;

$\vec{h}_s^g$ – вектор стану, нормалізований пакетною нормалізацією;

$\vec{o}_s^g$ – результат шару представлення.

Щоб спроектувати $\vec{o}_s^g$ у розподіл слів $\vec{d}_f$, підмережа містить лінійний шар і використовується шар softmax, а перетворення відбувається наступним чином:

$$\vec{d}_f = \text{softmax}\left(W_\omega^g \vec{o}_s^g + \vec{b}_\omega^g\right),$$

де $W_{\omega}^{g} \in \mathbb{R}^{V \times S}$ та $\vec{b}_{\omega}^{g}$ – вагова матриця та зміщення шару розподілу слів;

$\vec{d}_f$ – розподіл слів, що відповідає $\vec{\theta}_f$.

Для кожного $\upsilon \in \{1,2,...,V\}$ υ-а компонента d_f^{υ} є ймовірністю υ-го слова у підробленому документі $\vec{d}_f$.

Мережа Дискримінатора. Дискримінатор складається з трьох шарів: $(V+K)$-вимірний шар спільного розподілу, S-вимірний шар представлення та вихідний шар, як показано на правій панелі рисунка 2.4. Він отримує реальну пару розподілу $\vec{p}_r$ і fake-пару розподілу $\vec{p}_f$ як вхід ($D_{вх}$), а потім виводить $D_{вих}$ для ідентифікації джерел введення (fake чи справжніх). Тобто, більш високе значення $D_{вих}$ означає, що D більш схильний передбачати вхідні дані як реальні, і навпаки [9].

Висновки за розділом 2

В другому розділі було сформульовано постановку задачі тематичного моделювання, розглянуто питання підготовки текстових документів до подальшого аналізу, досліджено нейронну тематичну модель.

Основна увага була зосереджена на застосуванні для розв’язання задачі тематичного моделювання нейромережевої моделі BAT. Ця модель здатна давати покращені результати узгодженості тем, що свідчить про її ефективність у вирішенні проблем, пов’язаних з досліджуваним питанням.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Мова Python 3 як інструмент тематичного моделювання

Python 3, як високорівнева мова програмування, вже давно стала фундаментальним інструментом у сфері досліджень тематичного моделювання. Її популярність можна пояснити поєднанням наступних факторів: синтаксис Python підкреслює читабельність та виразність, дозволяючи розробникам писати ясний та короткий код; простота мови полегшує навчання, роблячи його доступним для новачків та надаючи потужні функції для досвідчених користувачів, відкритий вихідний код Python сприяє співпраці та розвитку спільноти. Універсальність Python очевидна з точки зору його застосування в різних областях: від веб-розробки та аналізу даних до штучного інтелекту і наукових обчислень. Широка доступність бібліотек та фреймворків, таких як NumPy, Pandas та TensorFlow, ще більше підвищує його привабливість, роблячи Python провідним вибором для дослідників та розробників у всьому світі.

У контексті тематичного моделювання Python 3 виділяється як цінний інструмент, що пропонує надійну систему, призначену для обробки природної мови (NLP) та задач машинного навчання. Такі бібліотеки, як NLTK (Natural Language Toolkit) та Gensim, надають спеціалізовані модулі для аналізу тексту та тематичного моделювання. Доступність розширених бібліотек, у тому числі Scikit-learn, спрощує реалізацію складних алгоритмів тематичного моделювання, дозволяючи дослідникам зосередитися на нюансах своєї роботи, а не вивчати тонкощі низькорівневого програмування. Підтримка об'єктно-орієнтованого програмування у Python покращує структурування коду, забезпечуючи модульність та повторне використання коду – ключові аспекти ітеративної розробки тематичних моделей.

Більше того, адаптованість Python 3 до різних форматів даних та можливості інтеграції з іншими технологіями роблять його ідеальною мовою для дослідницьких потреб. Повна інтеграція з базами даних та інструментами візуалізації забезпечує цілісний підхід до тематичного моделювання.

Універсальність мови дозволяє дослідникам ефективно маніпулювати даними та попередньо їх обробляти, що є фундаментальним кроком у конвеєрі тематичного моделювання. Активна спільнота Python і велика документація ще більше сприяють простоті розробки, надаючи безліч ресурсів дослідникам, які знаються на тонкощах тематичного моделювання.

Таким чином, Python 3 надає ідеальний інструментарій для виконання завдань тематичного моделювання, підтримуючи високий рівень гнучкості та ефективності для досліджень у даному напрямку.

3.2 Алгоритм розв’язання задачі тематичного моделювання наукових текстів за допомогою нейронних мереж

Побудуємо алгоритм розв’язання задачі тематичного моделювання. Можна виділити чотири основні етапи алгоритму.

На першому етапі проводиться збір текстових даних, які будуть досліджуватись в подальшому. В нашому випадку збирається колекція наукових текстових документів.

На другому етапі починається робота з даними, що ми отримали раніше, а саме, проводиться їх попередня обробка.

На третьому етапі виконуємо перетворення підготовлених текстових документів до векторного виду за допомогою методу «мішка слів».

Четвертим, фінальним, етапом є розв’язання задачі тематичного моделювання, тобто визначення тематик документів колекції із застосуванням нейронних мереж.

Отже, для розв’язання поставленої у кваліфікаційній роботі задачі необхідно виконати наступні дії.

Етап 1. Збір текстових даних наукової спрямованості:

1) реалізація коду веб-парсера для збору колекції наукової текстової інформації у відкритих інтернет-джерелах.

Етап 2. Попередня обробка даних:

1) розбиття текстів на слова;

2) видалення гіперпосилань, електронних адрес, дат, чисел, спеціальних символів тощо;

3) видалення стоп-слів;

4) стемінг, лематизація.

Етап 3. Перетворення даних до векторного виду:

1) визначення V – розміру словника унікальних слів;

2) формування словника унікальних слів за колекцією документів;

3) формування вектору «мішок слів» для кожного документу колекції.

Етап 4. Розв'язання задачі тематичного моделювання за допомогою моделі BAT:

1) реалізація коду для архітектури моделі нейронної мережі BAT, загрузки набору даних, тренування та тестування;

2) процес тренування (мережа генератору);

3) оцінка натренованої моделі на тестовому наборі;

Результатом роботи алгоритму будуть отриманні значення похибки E, G та D за допомогою яких ми зможемо проаналізувати роботу програми та правильність її відпрацювання. А також отриманні значення ймовірнісного розподілу тем колекції та унікальних слів за кожною з тем. Найбільш уживані слова за кожної темою представлено за допомогою хмар слів.

3.3 Опис програми

Програма для тематичного моделювання наукових текстів написана мовою Python версії 3.8.12. Для збору текстових даних використовувався веб-парсер, який шукав наукові статті за сімома темами в інтернет-бібліотеках наукової періодики. Зібрана інформація включала назву, авторів та анотацію статей.

Загалом програмний код містить процедури загрузки, обробки і зберігання набору даних, реалізацію моделі, ініціацію процесу навчання та оцінки якості моделі. Фреймворк було імплементовано за допомогою середи розробки PyCharm, віртуального середовища згенерованого за допомогою менеджеру пакетів conda та низки різноманітних бібліотек.

Програмна реалізація моделі BAT (Bidirectional Adversarial Topic Model) тематичного моделювання зроблена в окремому .py файлі у вигляді класу. В основу розробленого програмного продукту покладено модель «GAN+Encoder» [10]. Використовуються бібліотеки PyTorch для реалізації нейронних мереж та інструменти для обробки даних. Основні класи моделі включають Generator, Encoder та Discriminator. Важливі параметри моделі включають розмір словникового простору (bow_dim), кількість тем (n_topic), розмір прихованого шару (hid_dim), пристрій (device), та ім'я завдання (taskname). Модель навчається за допомогою оптимізаторів Adam для генератора, енкодера та дискримінатора. Тренування включає альтернативне навчання генератора та дискримінатора.

На кожній епосі виводяться середні втрати для кожного компонента моделі, а також тематичні слова для візуальної оцінки навчання. Крім того, код має функції для оцінки якості тем та відображення тематичних слів. Функції оцінки використовуються для оцінки різних метрик якості тематичного моделювання, таких як узгодженість (c_v, c_w2v), унікальність (c_uci, c_npmi), мінімізація тематичних збурень (mimno_tc) та дивергенція тем (td).

У коді також присутні функції для відображення тематичних слів та оцінки якості тем, що може бути важливим для подальшого аналізу та покращення моделі. Виведення результатів під час тренування, включаючи втрати та тематичні слова, робить код зручним для візуальної оцінки навчання.

Також важливим є реалізація архітектур модулів Generator, Encoder та Discriminator для моделі тематичного моделювання. Основні компоненти цих класів використовуються для генерації, енкодування та дискримінації тематичних представлень документів:

а) Generator (Генератор):

1) має лінійний блок для перетворення вхідного тематичного розподілу (theta) у вихідний розподіл слів (bow_f);

2) використовує лінійний шар та softmax-функцію для отримання ймовірностей слів;

б) Encoder (Енкодер):

1) використовує лінійний блок для перетворення вектора слів (bow) у тематичний розподіл (theta);

2) включає лінійний шар та softmax-функцію для отримання ймовірностей тем;

в) Discriminator (Дискримінатор):

1) комбінує тематичний розподіл (theta) та вектор слів (bow) для створення репрезентації документа;

2) використовує лінійний блок та sigmoid-функцію для оцінки ймовірності того, що документ є «реальним».

У всіх класах використовується блок з лінійним шаром, функцією активації LeakyReLU та, за необхідності, шаром нормалізації. Це допомагає стабілізувати та поліпшити навчання моделі. Кожен клас також містить метод forward для визначення проходження даних через модель.

Для створення та обробки набору даних використовується клас DocDataset. Клас включає методи для токенізації текстів, побудови словника, перетворення текстів у мішок слів (BOW), а також можливість використання TF-IDF. Набір даних може бути використаний для тренування тематичної моделі на основі BATM.

Основні компоненти класу:

а) ініціалізація та побудова словника:

1) здійснює зчитування текстів з файлу, токенізацію, та побудову словника;

2) підтримує використання стоп-слів, фільтрацію за нижньою та верхньою межами, а також можливість використання TF-IDF;

3) використовує бібліотеку Gensim для роботи з текстовими даними та побудови словника;

б) представлення даних у форматі BOW та TF-IDF:

1) тексти конвертуються у вектори, які представляють собою мішки слів (BOW);

2) є можливість використання TF-IDF для представлення текстів;

в) серіалізація та відновлення даних – дані можуть бути збережені та відновлені для подальшого використання;

г) функції для виводу статистики – методи для виводу найчастіших токенів з найвищим та найнижчим document frequency (DF) та collection frequency (CF);

ґ) функції для отримання даних – методи для отримання елементів набору даних та їх довжини.

Клас також реалізує ітератор для зручного перегляду текстових даних, та взагалі, даний клас є важливою частиною підготовки даних для моделі тематичного моделювання та надає зручний інтерфейс для роботи з текстовими даними.

Під час обчислювальних експериментів ми дослідили розв’язання задачі тематичного моделювання для кількостей топіків 6, 7, 8 та 12. Для кожного випадку було навчено модель і отримані результати були представлені у вигляді хмар слів і списків ключових слів у порядку спадання їх частот.

Скрипт ініціації процесу навчання використовує параметри командного рядка для конфігурації навчання. Нижче наведено опис кожного параметра:

– taskname – ідентифікатор завдання, яке використовується для збереження результатів та конструювання шляхів до файлів;

– no_below – нижня межа кількості зустрічей слів у тексті; слова, які зустрічаються менше разу, виключаються з аналізу;

– no_above – верхня межа відносної кількості зустрічей слів у тексті; слова, які зустрічаються в 100% текстів, виключаються з аналізу;

– num_epochs – кількість епох навчання моделі;

– n_topic – кількість тем, які модель намагається виділити в текстах;

– bkpt_continue – перемикач, який вказує, чи потрібно завантажити попередньо навчену модель та продовжити навчання;

– use_tfidf – перемикач, який вказує, чи використовувати TF-IDF для векторизації текстів;

– rebuild – перемикач, який вказує, чи потрібно відновлювати корпус (наприклад, знову токенізувати текст, побудувати словник і т.д.);

– dist – розподіл, використовуваний для latent vectors: 'dirichlet', 'gmm_std', 'gmm_ctm', 'gaussian' і т.д;

– batch_size – розмір пакету даних для одного кроку оптимізації;

– criterion – критерій для обчислення втрат: «cross_entropy», «bce_softmax», «bce_sigmoid»;

– auto_adj – перемикач, що вказує, чи слід автоматично налаштовувати параметр no_above (зниження верхньої межі відносної кількості зустрічей слів у тексті);

– lang – мова датасету, впливає на вибір токенізатора та інші параметри.

Скрипт завантажує датасет, ініціалізує модель з вказаними параметрами та запускає процес навчання. Після навчання відбувається оцінка моделі та збереження її ваг.

Програма зручна у використанні завдяки своїй чіткій файловій структурі та коду, написаному відповідно до стандарту PEP 8.

Висновки за розділом 3

В розділі 3 представлений опис програми, написаної мовою Python версії 3.8.12. Було зроблено кілька важливих кроків для обробки та підготовки тестових даних для подальшого аналізу. Застосування вебпарсеру полегшило збір тестових даних, після якого було проведено попередню обробку даних для оптимізації подальших завдань, пов’язаних із текстом. Наступним кроком стала

трансформація документів у векторний вид за допомогою методу «мішок слів», що дало змогу отримати підготовлену колекцію документів, яка використовується для взаємодії з розробленим програмним забезпеченням.

Програмна реалізація BAT моделі для тематичного моделювання була реалізована в окремому файлі .py, організованому як клас. Бібліотеки PyTorch разом із інструментами обробки даних відіграли важливу роль у впровадженні нейронних мереж. Основні класи моделі охоплювали генератор, енкодер і дискримінатор. Навчання включало оптимізатори Adam.

Код також містить функції для візуалізації тематичних слів і оцінки якості теми, сприяючи подальшому аналізу та покращенню моделі. Виведення результатів навчання в реальному часі, включаючи втрати та тематичні слова, покращило зручність використання коду для візуальної оцінки навчання.

Акцент було зроблено на реалізації архітектур модулів Generator, Encoder і Discriminator для моделі тематичного моделювання. Ключові компоненти цих класів були використані для створення, кодування та розрізнення тематичних представлень документів.

4 РЕЗУЛЬТАТИ ОБЧИСЛЮВАЛЬНОГО ЕКСПЕРИМЕНТУ ТА ЇХ АНАЛІЗ

Розглянемо результати експерименту, де використовувався алгоритм для аналізу тематики наукових текстів. Описаний у розділі 3 програмний продукт було використано для визначення ймовірнісного розподілу тем у колекції документів та виділення найхарактерніших термінів для кожної теми.

Для роботи було зібрано колекцію наукових документів обсягом 4608 документів за такими темами: актуарна математика (actuar_math), кластерний аналіз (cluster_analysis), електродинаміка (electrodynamic), інвестування (investing), математична фізика (math_phys), нейронні мережі (neural_network), оптимальне керування (optimal_control).

Задля підготовки набору даних до використання при навчанні моделі було проведено попередню обробку назв та анотацій документів, а саме: стемінг та лематизація, видалення коротких документів, гіперпосилань, стоп-слів тощо. Після попередньої обробки та чищення маємо набір з 3630 текстів, які на наступному кроці токенізуються за допомогою токенізатору SpaCy згідно з обраною мовою. Далі на основі токенізованого тексту створюється словник, він фільтрується за допомого деяких параметрів, потім це переводиться в «bag of words» представлення та застосовується метод TF-IDF, а саме: спочатку обчислюється міра TF-IDF для кожного документа, розраховується модель TF-IDF на основі «bag of words» представлення тексту, застосовується ця модель до кожного документа, отримуючи новий TF-IDF представлення, враховується частота входження терміну у документ та обернена частота входження терміну у всі документи. Потім деякі обчислені дані серіалізуються у тимчасові файли для подальшого використання та прискорення завантаження датасету при майбутніх запусках.

Тимчасові файли включають:

– corpus.mm – «bag of words» представлення тексту;

– tfidf.mm – TF-IDF представлення;

– dict.txt – серіалізований словник;

– docs.pkl – серіалізований список токенізованих документів.

Після обробки документів застосовується навчання моделі BAT для розв'язання задачі тематичного моделювання для випадків прихованих топіків. Для кожного з експериментів були отримані наступні результати:

а) побудовані графіки значень помилок для кожного з модулів, де похибка генератора буде зображена на графіках блакитним кольором та мати назву loss_G, похибка дискримінатора буде зображена помаранчевим кольором та мати назву loss_D, похибка енкодера буде зображена зеленим кольором та мати назву loss_E (рис. 3.2, 3.4, 3.6, 3.8 та 3.10);

б) прораховані такі метрики як td_score (різноманітність тем), c_v (оцінка згуртованості), c_uci (оцінка злагодженості), c_npmi (оцінка нормалізованої точкової взаємної інформації), mimno_tc (оцінка узгодженості тем Mimno) (таблиці 3.1 – 3.5);

в) згенеровані хмари слів, що візуалізують отримані ймовірнісні розподіли слів за топіками (рис. 3.1, 3.3, 3.5, 3.7 та 3.9);

г) виведені найчастіше повторювані слова та їх ймовірнісний розподіл для кожного з топіків (таблиці 3.6 – 3.10);

ґ) виведено ймовірнісний розподіл окремих документів з тем (таблиця 3.11 – результати для випадку 7 топіків при 50 епохах).

Було зроблено 5 експериментів:

а) кількість топіків – 7, кількість епох – 300;

б) кількість топіків – 7, кількість епох – 50;

в) кількість топіків – 6, кількість епох – 50;

г) кількість топіків – 8, кількість епох – 50;

д) кількість топіків – 12, кількість епох – 50.

Експеримент 1.

Для аналізу похибок генератора, енкодера та дискримінатора зобразимо їх на рис. 3.1.

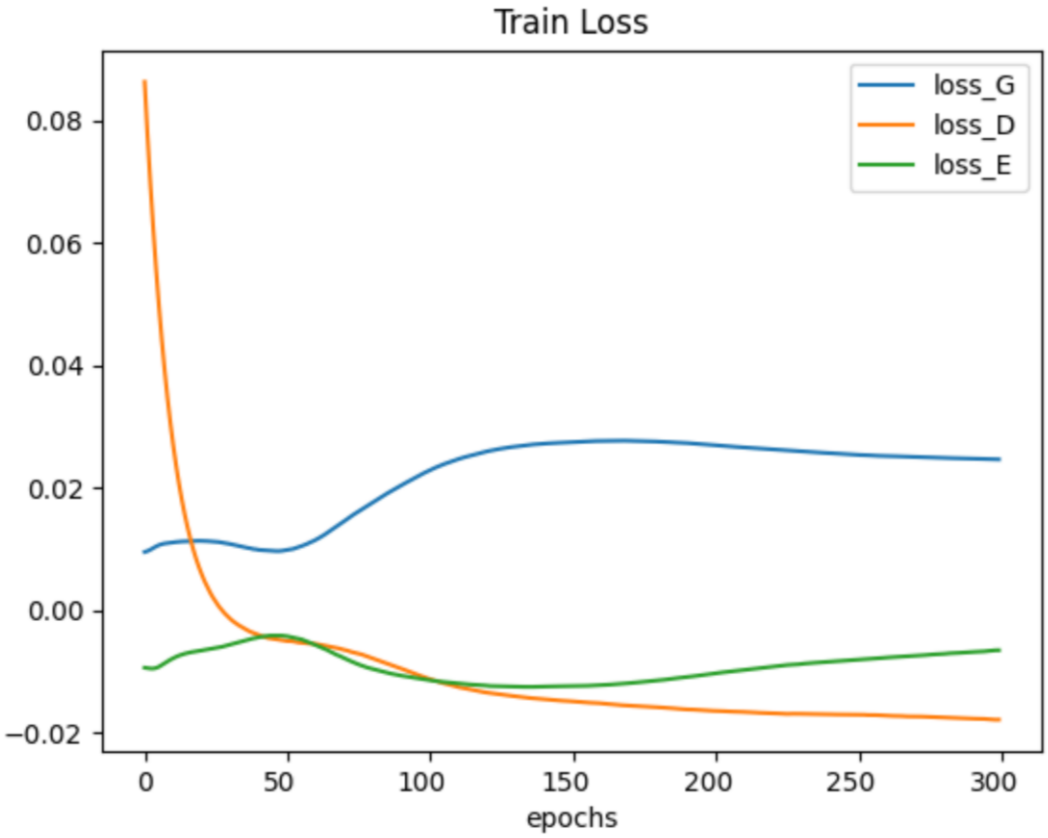


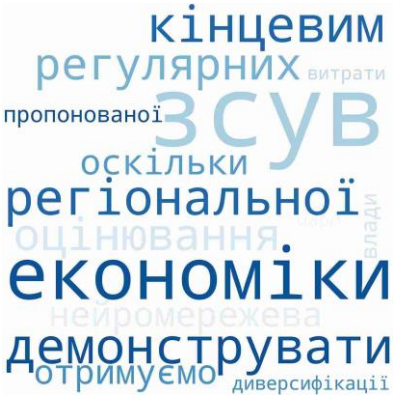
Рисунок 3.1 – Тренувальна похибка для кожного з модулів (Generator, Discriminator, Encoder) для 7 топіків при 300 епохах

Можемо бачити, що тільки помилка дискримінатора впала спочатку тренування, при тому, що помилка генератора збільшилась, а енкодера залишилась приблизно такою ж, якою і була на початку. Така ситуація свідчить про те, що дискримінатор покращив свою можливість відрізняти реальні тексти від згенерованих, при тому, що генератор погіршив навички до генерації більш реальних текстів, а енкодер практично ніяк не змінив свої якості в діставанні ознак з текстів.

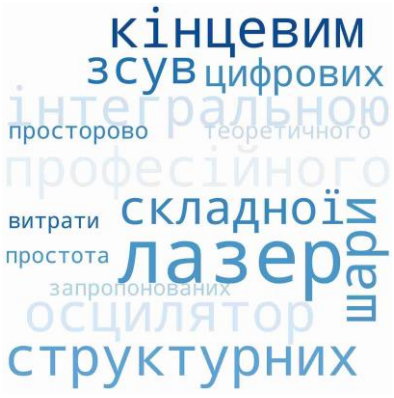
В таблиці 3.1. зобразимо значення метрик на останній епосі, та зробимо візуалізуємо отримані ключові слова за допомогою хмари слів (рис.3.2).

Таблиця 3.1 – Значення метрик на останній епосі, 7 топіків, 3000 епох

td	c_v	c_uci	c_npmi	mimno_tc
0,66	0,64	–12,74	–0,45	–214,2



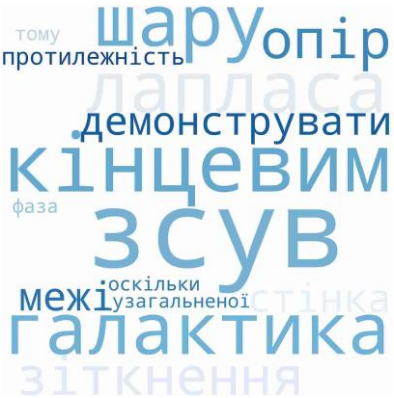
а) Топік 1



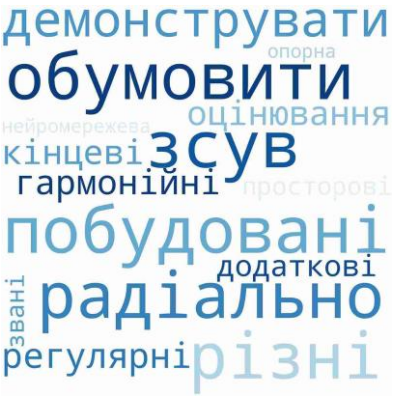
б) Топік 2

в) Топік 3

г) Топік 4



д) Топік 5



е) Топік 6

ж) Топік 7

Рисунок 3.2 – Хмари слів для ймовірнісного розподілу
слів для 7 топіків при 300 епохах

Експеримент 2.

Для аналізу похибок генератора, енкодера та дискримінатора зобразимо їх на рис.3.3.

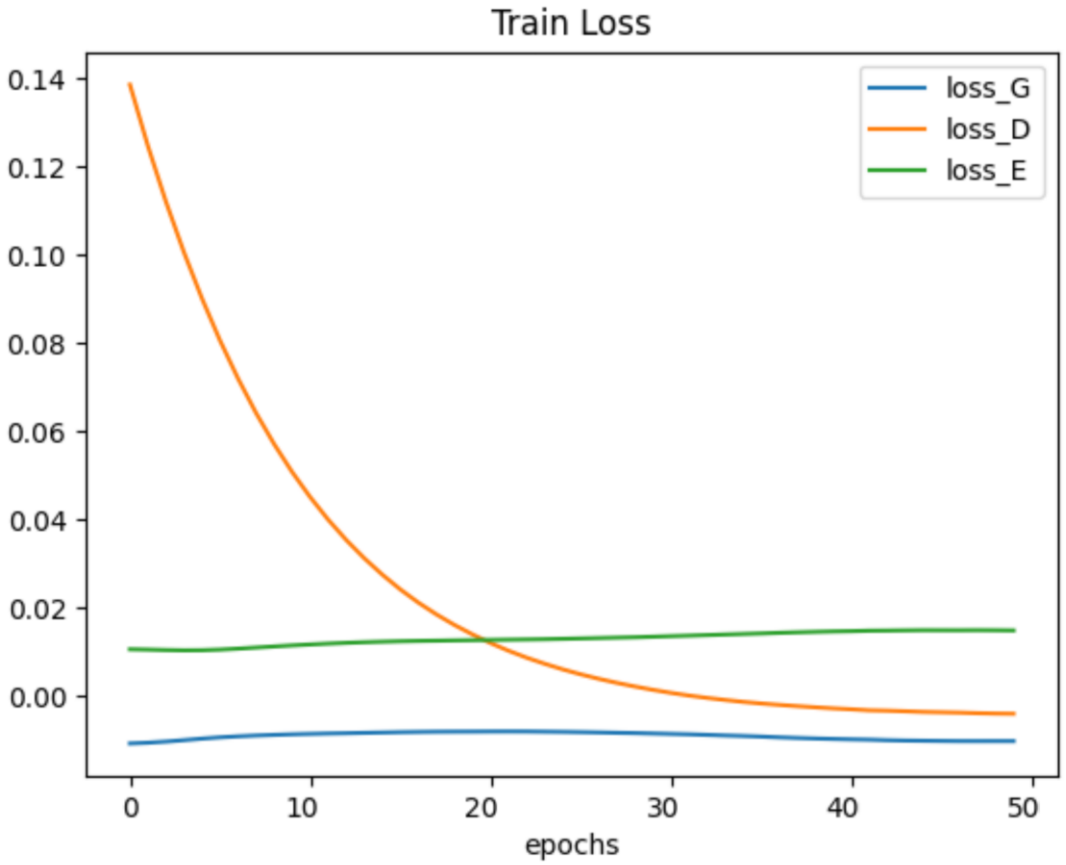


Рисунок 3.3 – Тренувальна похибка для кожного з модулів (Generator, Discriminator, Encoder) для 7 топіків при 50 епохах

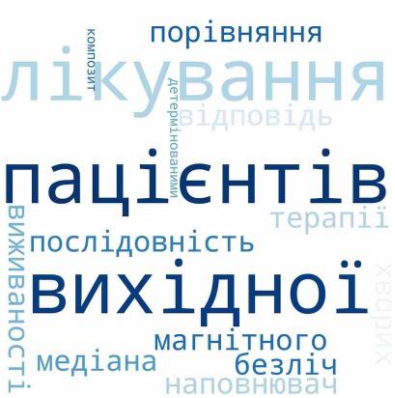
З рисунку 3.3 можемо бачити, що тільки помилка дискримінатора впала спочатку тренування, при тому, що помилка генератору не змінилась, а кодувальника залишилась приблизно такою ж, якою і була на початку. Така ситуація свідчить про те, що дискримінатор покращив свою змогу відрізняти реальні тексти від згенерованих, при тому, що генератор не змінив свої навички до генерації більш реальних текстів, та кодувальник практично ніяк не змінив свої якості в діставанні ознак з текстів.

Таблиця 3.2 – Значення метрик на останній епосі, 7 топіків, 50 епох

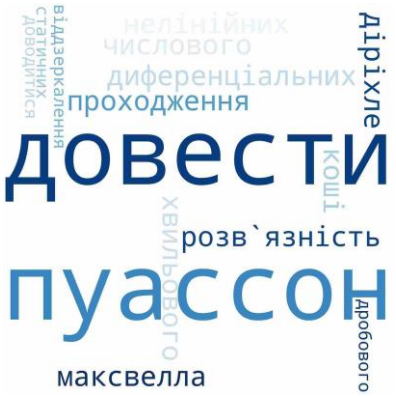
td	c_v	c_uci	c_npmi	mimno_tc
1	0,47	–8,67	–0,27	–217,14

а) Топік 1

б) Топік 2

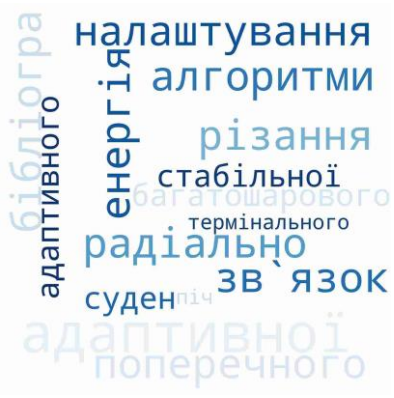


в) Топік 3



г) Топік 4

д) Топік 5



е) Топік 6

ж) Топік 7

Рисунок 3.4 – Хмари слів для ймовірнісного розподілу слів для 7 топіків при 50 епохах

Експеримент 3.

Для аналізу похибок генератора, енкодера та дискримінатора зобразимо їх на рис.3.5.

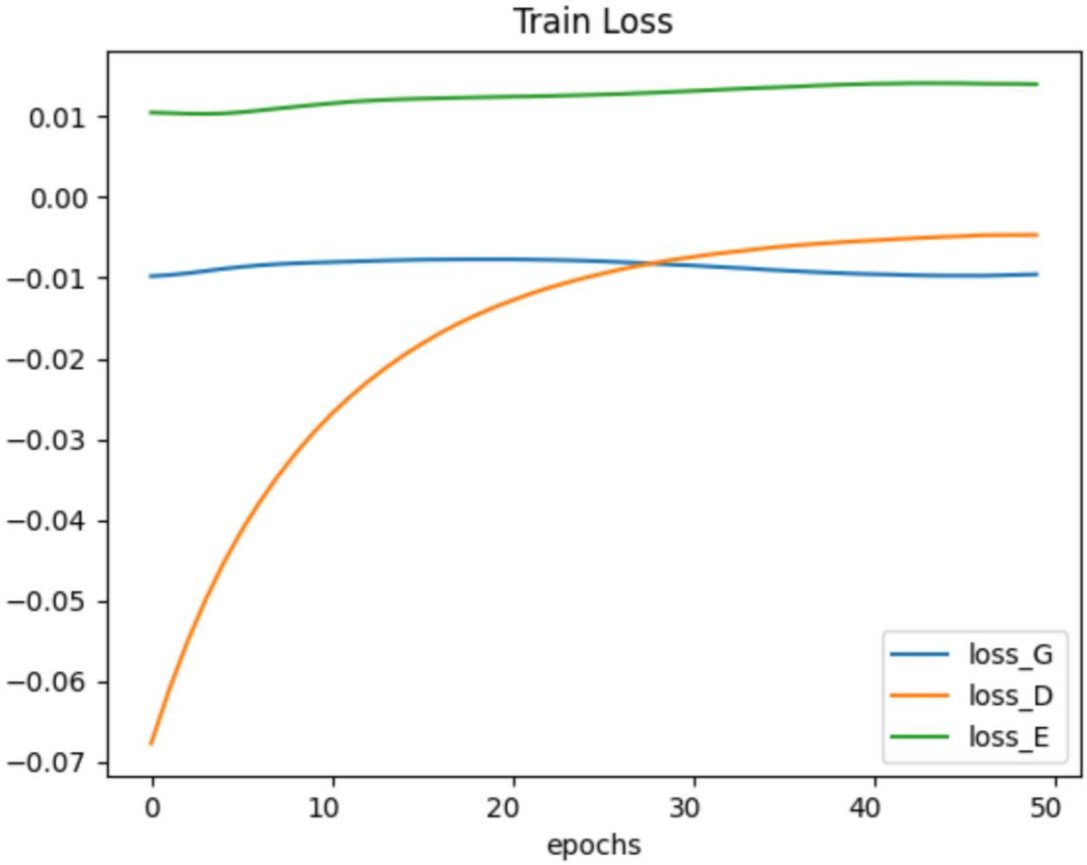
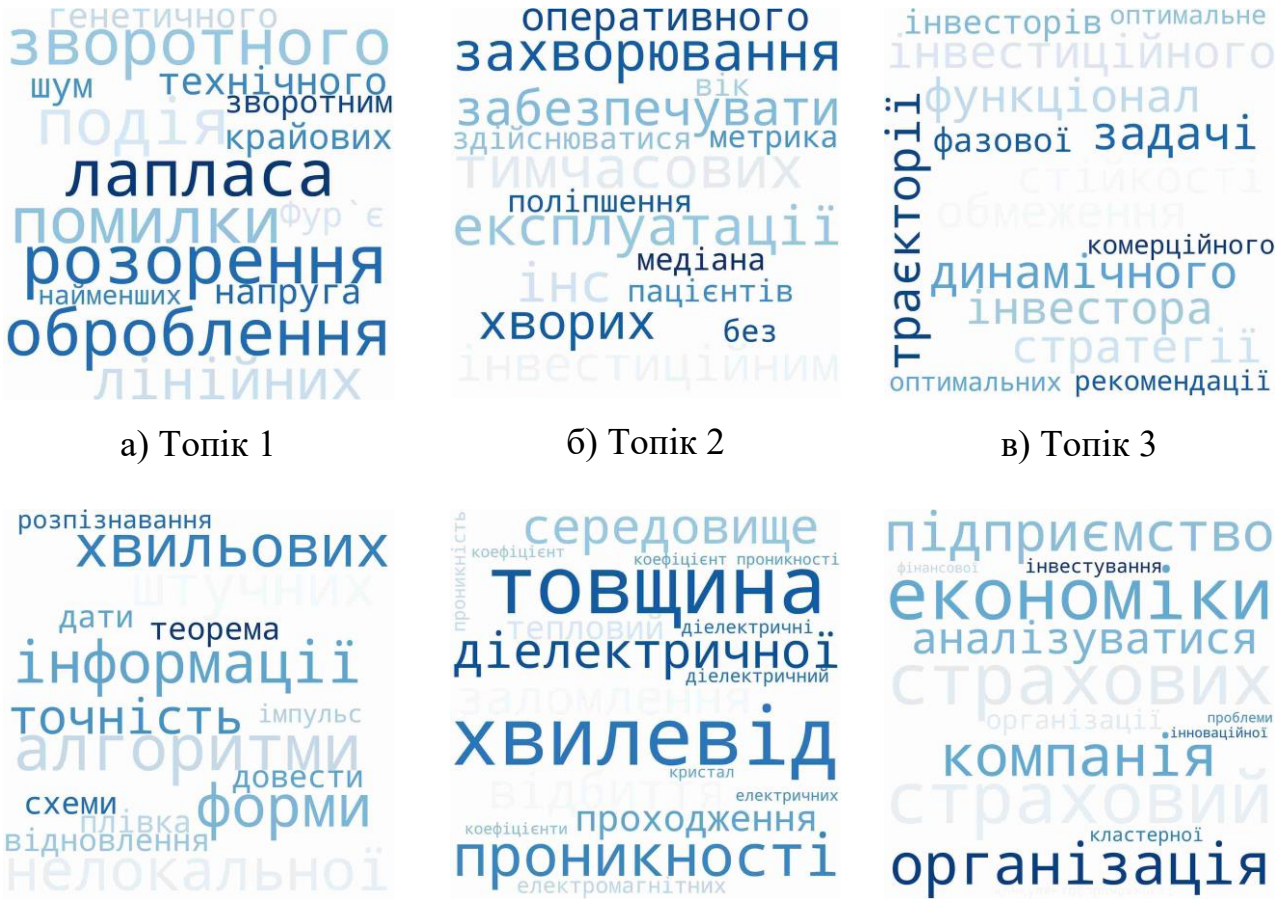


Рисунок 3.5 – Тренувальна похибка для кожного з модулів (Generator, Discriminator, Encoder) для 6 топіків при 50 епохах

З рисунку 3.5 можемо бачити, що помилка дискримінатора зросла спочатку тренування, при тому, що помилка генератору і кодувальника залишилась приблизно такою ж, якою і була на початку. Така ситуація свідчить про те, що дискримінатор погіршив свою змогу відрізняти реальні тексти від згенерованих, при тому, що генератор не змінив свої навички до генерації більш реальних текстів, та кодувальник практично ніяк не змінив свої якості в діставанні ознак з текстів.

Таблиця 3.3 – Значення метрик на останній епосі, 6 топіків, 50 епох

td	c_v	c_uci	c_npmi	mimno_tc
1	0,59	–7,2	–0,21	–253



а) Топік 1 б) Топік 2 в) Топік 3

г) Топік 4 д) Топік 5 е) Топік 6

Рисунок 3.6 – Хмари слів для ймовірнісного розподілу слів для 6 топіків при 50 епохах

Експеримент 4.

Для аналізу похибок генератора, енкодера та дискримінатора зобразимо їх на рис.3.7.

З рисунку 3.7 можемо бачити, що помилка дискримінатора зросла спочатку тренування, при тому, що помилка генератору і кодувальника залишилась приблизно такою ж, якою і була на початку. Така ситуація свідчить про те, що дискримінатор погіршив свою змогу відрізняти реальні тексти від згенерованих, при тому, що генератор не змінив свої навички до генерації більш реальних текстів, та кодувальник практично ніяк не змінив свої якості в діставанні ознак з текстів.

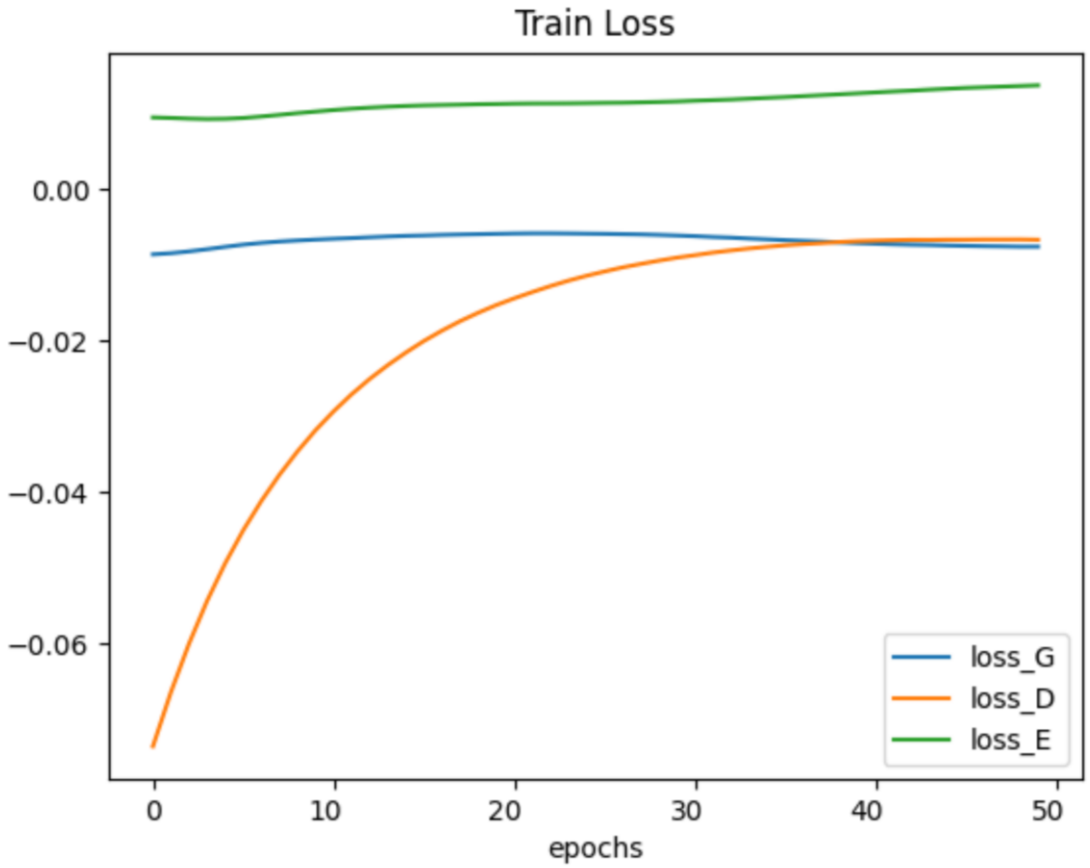


Рисунок 3.7 – Тренувальна похибка для кожного з модулів (Generator, Discriminator, Encoder) для 8 топіків при 50 епохах

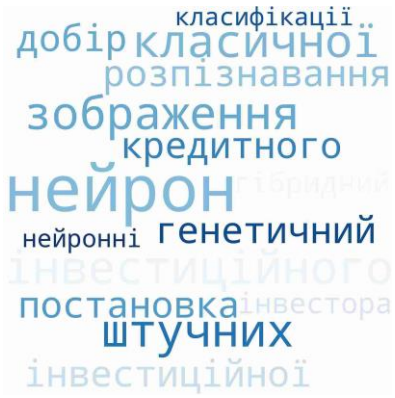
Таблиця 3.4 – Значення метрик на останній епосі, 8 топіків, 50 епох

td	c_v	c_uci	c_npmi	mimno_tc
1	0,46	–9,77	–0,31	–208,12

Експеримент 5.

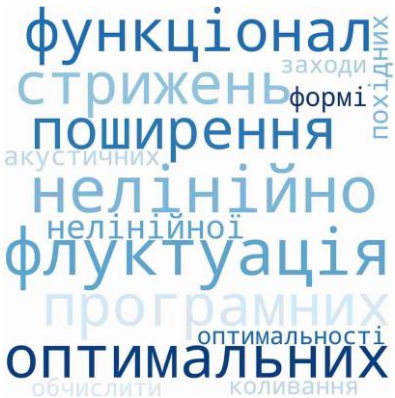
Для аналізу похибок генератора, енкодера та дискримінатора зобразимо їх на рис. 3.9.

З рисунку 3.9 можемо бачити, що тільки помилка дискримінатора впала спочатку тренування, при тому, що помилка генератору не змінилась, а кодувальника збільшилась. Така ситуація свідчить про те, що дискримінатор покращив свою змогу відрізняти реальні тексти від згенерованих, при тому, що генератор не змінив свої навички до генерації більш реальних текстів, а кодувальник погіршив свої якості в діставанні ознак з текстів.

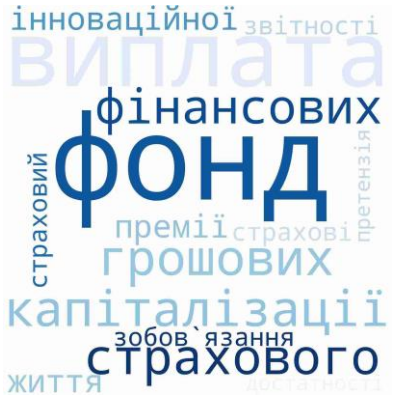


а) Топік 1

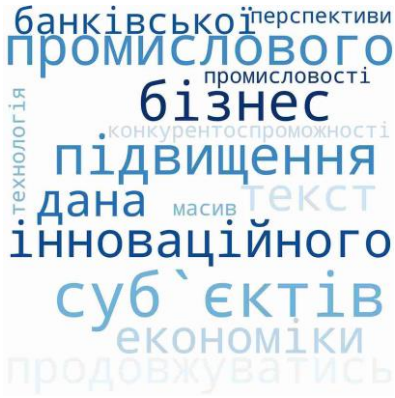
б) Топік 2



в) Топік 3

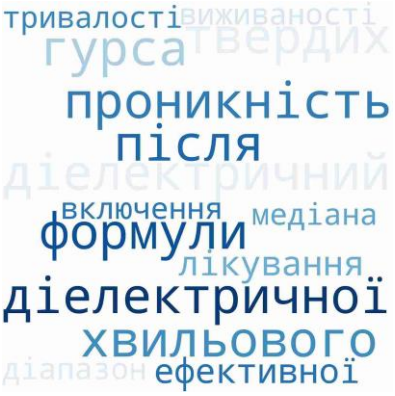


г) Топік 4

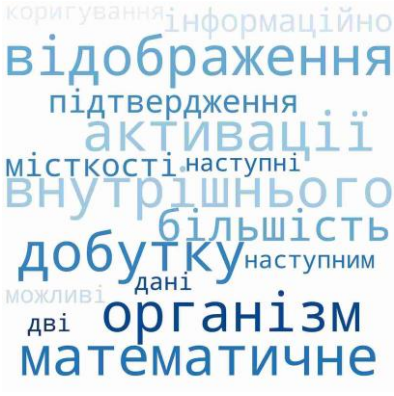


д) Топік 5

е) Топік 6



є) Топік 7



ж) Топік 8

Рисунок 3.8 – Хмари слів для ймовірнісного розподілу слів для 8 топіків при 50 епохах

Аналізуючи отримані результати ти виходячи зі значень метрик E, G та D можна сказати, що з перших двох експериментів, де кількість топіків на виході дорівнює кількості топіків на вході, другий експеримент був найкращим, бо високі значення метрик показують більшу якість моделі, тобто розподіл по 7

топікам з навчанням на 50 епох є більш відповідним ніж при розподілі на 300 епох. Таке може статися через перенавчання моделі, адже на рис.2.1 ми бачимо як відбувається значне підвищення похибки генератора G одразу після 50 епох.

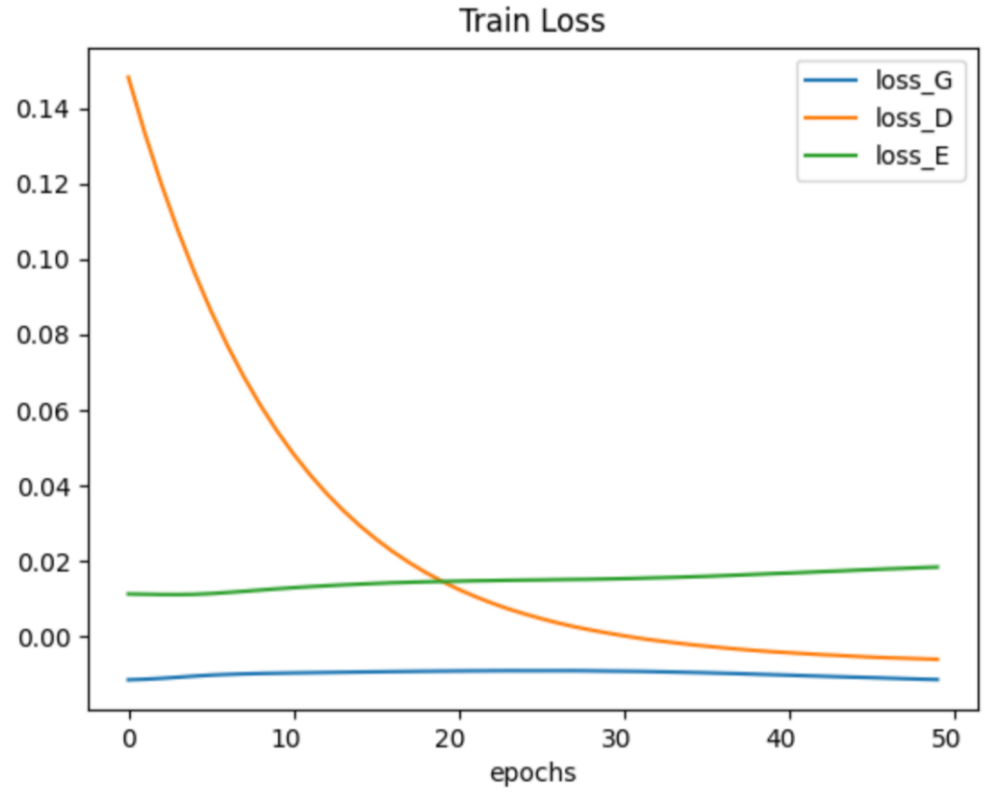
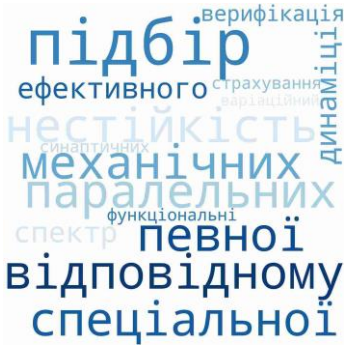


Рисунок 3.9 – Тренувальна похибка для кожного з модулів (Generator, Discriminator, Encoder) для 12 топіків при 50 епохах

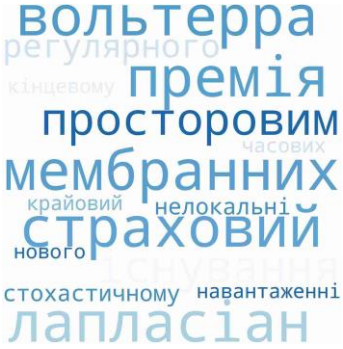
Таблиця 3.5 – Значення метрик на останній епосі, 12 топіків, 50 епох

td	c_v	c_uci	c_npmi	mimno_tc
0.99	0,56	–11,19	–0,38	–172,68

Проаналізувавши значення метрик різноманітності тем та оцінки згрупованості, оцінки злагодженості, можна прийти до висновку, що розподіл на 7 топіків при 50 епохах буде мати найкращий показник злагодженості тем. Другим по якості буде йти розподіл датасету на 8 топіків і на 12. Це зумовлено тим, що при більшому розбитті тем програма має змогу краще виділити теми виходячи з їх ключових слів.

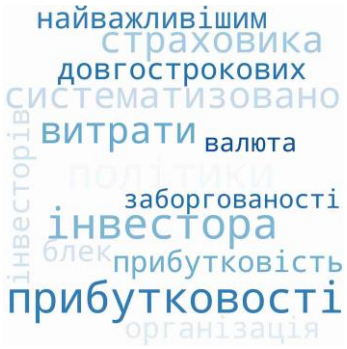


а) Топік 1

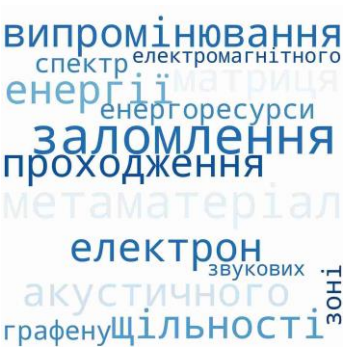


б) Топік 2

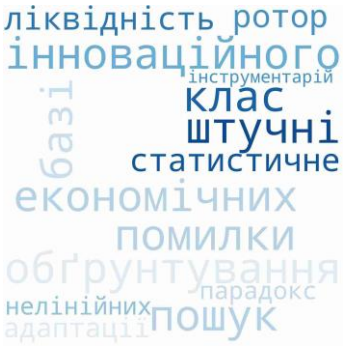
в) Топік 3



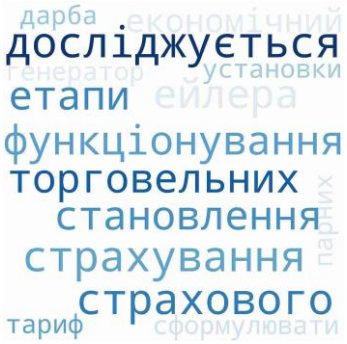
г) Топік 4



д) Топік 5

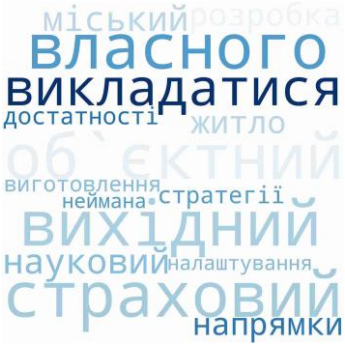


е) Топік 6



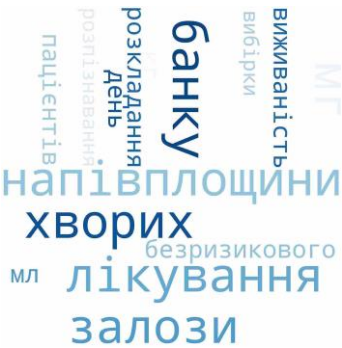
є) Топік 7

ж) Топік 8

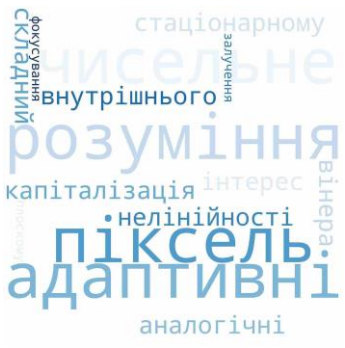


з) Топік 9

и) Топік 10



і) Топік 11



й) Топік 12

Рисунок 3.10 – Хмари слів для ймовірнісного розподілу слів для 12 топіків при 50 епохах

Для глибшого аналізу саме розподілу слів за топіками та їх частоти використання в таблицях 3.6 – 3.10 наведемо ймовірнісний розподіл слів за темами для п’ятнадцяти найуживаніших слів в топіку. Справа від слова будемо позначати його частоту появи.

Аналізуючи отримані результати у таблицях 3.6 – 3.10, можна зробити висновок щодо тематичної спрямованості кожного окремого топіка, використовуючи його найуживаніші слова. Важливо відзначити, що деякі зі слів, визначених алгоритмом як ключові, можуть входити в кілька топіків або є загальновживаними. З метою поліпшення результатів тематичного моделювання доцільно виключати такі слова з датасету, але для цього необхідно буде провести його додаткове очищення.

Також зазначимо, що основний акцент був зроблений на використанні анотацій, а не повних наукових текстів, що суттєво вплинуло на результати моделювання та окремі набори ключових слів, виділених алгоритмом топіків. Варто зауважити, що аналіз повних текстів може надати більш чітку інформацію щодо тематичної спрямованості, однак, навіть при обмеженні анотаціями, ключові слова добре відображають тематику топіків і залишаються ефективним інструментом для визначення тематичних спрямувань у вивчених статтях. З цим урахуванням можна подальше дослідження направити на розширення датасету чи додатковий аналіз повних текстів для отримання більш глибокого розуміння тем та їх представлення у наукових дослідженнях.

Проаналізуємо таблицю 3.7, яка містить ймовірнісні розподіли слів за топіками для випадку 7 топіків при 50 епохах (виходячи з порівняльного аналізу рисунків 3.1 та 3.2 ми вже дійшли до висновку, що навчання на 50 епохах було більш точним, ніж на 300). Оскільки цей випадок відповідає реальній кількості тем, за якими було зібрано документи у датасеті, з легкістю порівняємо, наскільки отримані набори ключових слів можна віднести до того чи іншого оригінальних топіків.

Таблиця 3.6 – Ймовірнісний розподіл слів за темами, $t = 6$ при 50 епохах

Topic 1:	Topic 2:	Topic 3:	Topic 4:	Topic 5:	Topic 6:
розорення (0.0078)	часових (0.0068)	інвестиційного (0.0081)	алгоритми (0.0067)	проникності (0.0151)	економіки (0.0072)
зворотного (0.0068)	захворювання (0.0048)	траєкторії (0.0080)	нелокальної (0.0065)	діелектричної (0.0120)	економічного (0.0059)
Лапласа (0.0059)	ІМН (0.0047)	динамічного (0.0044)	форми (0.0053)	середа (0.0074)	страхових (0.0057)
подія (0.0055)	експлуатації (0.0045)	обмеження (0.0038)	штучних (0.0044)	відбиття (0.0067)	страхова (0.0054)
обробки (0.0054)	інвестиційним (0,0043)	задача (0.0036)	хвильових (0.0037)	заломлення (0.0065)	організація (0.0050)
помилки (0.0051)	забезпечувати (0.0038)	стратегії (0.0033)	інформації (0.0037)	проходження (0.0065)	компанія (0.0047)
лінійних (0.0044)	хворих (0.0035)	функціонал (0.0031)	точність (0.0034)	хвилевід (0.0064)	підприємство (0.0047)
технічного (0.0044)	оперативного (0.0034)	стійкості (0.0030)	відновлення (0.0034)	теплової (0.0059)	аналізується (0.0041)
крайових (0.0043)	здійснюватися (0.0034)	інвестора (0.0029)	теорема (0.0032)	електромагнітних (0.0055)	організації (0.0039)
напруга (0.0041)	пацієнтів (0.0034)	інвесторів (0.0029)	плівка (0.0030)	діелектричний (0.0053)	інвестування (0.0038)
генетичного (0.0034)	вік (0.0029)	фазової (0.0028)	схеми (0.0029)	проникність (0.0052)	кластерної (0.0037)
шум (0.0034)	без (0.0028)	рекомендації (0.0027)	довести (0.0029)	товщини (0.0051)	конкурентоспроможності (0.0037)
Фур'є (0.0033)	медіани (0.0028)	комерційного (0.0027)	дати (0.0027)	діелектричних (0.0050)	фінансової (0.0035)
зворотнім (0.0032)	покращення (0.0028)	оптимальних (0.0026)	розпізнавання (0.0027)	коефіцієнти (0.0049)	проблеми (0.0034)
найменших (0.0030)	метрика (0.0028)	оптимальне (0.0025)	імпульс (0.0026)	кристал (0.0048)	інноваційної (0.0031)

Таблиця 3.7 – Ймовірнісний розподіл слів за темами, $t = 7$ при 50 епохах

Topic 1:	Topic 2:	Topic 3:	Topic 4:	Topic 5:	Topic 6:	Topic 7:
фонд (0.0137)	магнітної (0.0070)	пацієнтів (0.0090)	довести (0.0065)	нейрон (0.0071)	радіально (0.0150)	інвестиційної (0.0170)
тепловий (0.0098)	сигнал (0.0064)	вихідної (0.0077)	пуассона (0.0057)	комплекс (0.0058)	різання (0.0147)	доходності (0.0052)
газ (0.0084)	пристрій (0.0057)	лікування (0.0065)	хвильового (0.0050)	промислового (0.0040)	бібліографія (0.0089)	страхової (0.0052)
температура (0.0081)	меж (0.0045)	виживання (0.0045)	диференціальних (0.0048)	підприємства (0.0036)	алгоритми (0.0087)	інвесторів (0.0051)
зобов'язаний (0.0060)	поглиблення (0.0040)	відповідь (0.0040)	коші (0.0045)	класифікація (0.0028)	адаптивної (0.0082)	інвестиційних (0.0039)
грошових (0.0060)	matlab (0.0038)	терапії (0.0040)	діріхле (0.0045)	регіон (0.0028)	настройки (0.0074)	дохід (0.0038)
баланс (0.0048)	акустичного (0.0037)	наповнювач (0.0039)	проходження (0.0040)	ресурс (0.0027)	енергія (0.0057)	фінансової (0.0031)
автомобіль (0.0047)	упругий (0.0036)	хворих (0.0039)	максвелла (0.0039)	нечіткої (0.0025)	зв'язок (0.0056)	бізнес (0.0028)
банкротства (0.0045)	графену (0.0035)	множина (0.0038)	нелінійних (0.0038)	переваги (0.0024)	поперечного (0.0056)	інструмент (0.0027)
потоків (0.0043)	інформація (0.0034)	порівняння (0.0038)	розв'язність (0.0038)	підприємство (0.0024)	багатошарової (0.0054)	фінансовий (0.0027)
характеристики (0.0039)	лагранж (0.0032)	послідовність (0.0036)	числового (0.0037)	марковица (0.0023)	судів (0.0052)	фінансових (0.0027)
теплоносій (0.0039)	зменшення (0.0032)	медіана (0.0034)	відбиття (0.0036)	функціонування (0.0022)	стабільної (0.0049)	цінні (0.0027)
суміші (0.0039)	антена (0.0029)	магнітного (0.0032)	статичних (0.0035)	захворювання (0.0021)	адаптивного (0.0047)	ризиками (0.0025)
гц (0.0036)	нм (0.0029)	визначеними (0.0031)	доводитися (0.0031)	промисловості (0.0021)	термінального (0.0045)	інвестиції (0.0025)
мембрани (0.0036)	де (0.0027)	композит (0.0031)	дробового (0.0030)	організм (0.0020)	піч (0.0043)	страхових (0.0024)

Таблиця 3.8 – Ймовірнісний розподіл слів за темами, $t = 7$ при 300 епохах

Topic 1:	Topic 2:	Topic 3:	Topic 4:	Topic 5:	Topic 6:	Topic 7:
зсув (0.0274)	лазер (0.0210)	становище (0.0168)	кошти (0.0305)	зсув (0.0126)	збудовані (0.0162)	var (0.0247)
економіки (0.0093)	професійного (0.0114)	різні (0.0118)	поширення (0.0182)	кінцевим (0.0106)	обумовити (0.0114)	зсув (0.0125)
демонструвати (0.0072)	осцилятор (0.0097)	ціноутворення (0.0097)	змінний (0.0163)	галактика (0.0094)	зсув (0.0104)	обумовити (0.0122)
кінцевим (0.0064)	інтегральної (0.0095)	зсув (0.0092)	просторово (0.0127)	шару (0.0091)	радіально (0.0096)	гібридної (0.0089)
регіональної (0.0063)	кінцевим (0.0092)	рівномірно (0.0072)	ін (0.0086)	лапласа (0.0081)	різні (0.0079)	опір (0.0083)
оцінювання (0.0059)	структурних (0.0087)	вартість (0.0064)	ракети (0.0079)	опір (0.0076)	продемонструвати (0.0077)	кошти (0.0073)
регулярних (0.0054)	шари (0.0084)	квадрат (0.0059)	теоретичного (0.0076)	зіткнення (0.0061)	регулярних (0.0077)	зіткнення (0.0067)
нейромережева (0.0052)	складною (0.0082)	регулярних (0.0058)	працювати (0.0073)	демонструвати (0.0057)	гармонійної (0.0072)	різниця (0.0066)
оскільки (0.0051)	зсув (0.0079)	додатковий (0.0057)	становище (0.0071)	кордону (0.0057)	оцінювання (0.0071)	перебування (0.0063)
отримуємо (0.0049)	цифрових (0.0075)	при (0.0056)	перебування (0.0063)	стінка (0.0055)	кінцевим (0.0064)	інших (0.0057)
пропонованої (0.0048)	теоретичного (0.0064)	розкривати (0.0055)	опір (0.0062)	протилежність (0.0055)	додатковий (0.0063)	електромагнітний (0.0057)
влади (0.0047)	просторово (0.0064)	експертних (0.0052)	інтегральної (0.0062)	узагальненої (0.0052)	просторово (0.0060)	гармонійної (0.0056)
диверсифікації (0.0044)	запропонованих (0.0063)	перебування (0.0051)	нейромережева (0.0061)	фаза (0.0049)	званих (0.0059)	просторово (0.0055)
витрати (0.0044)	простота (0.0062)	фонди (0.0049)	крайових (0.0059)	тому (0.0045)	нейромережева (0.0059)	реакція (0.0055)
шари (0.0042)	витрати (0.0060)	диверсифікації (0.0048)	гармонійної (0.0059)	оскільки (0.0045)	опорний (0.0057)	ін (0.0053)

Таблиця 3.9 – Ймовірнісний розподіл слів за темами, $t = 8$ при 50 епохах

Topic 1:	Topic 2:	Topic 3:	Topic 4:	Topic 5:	Topic 6:	Topic 7:	Topic 8:
нейрон (0.0053)	дарба (0.0112)	флуктуація (0.0107)	фонд (0.0089)	суб'єктів (0.0225)	активності (0.0080)	діелектричної (0.0112)	внутрішнього (0.0105)
інвестиційного (0.0053)	ейлера (0.0056)	нелінійно (0.0071)	виплата (0.0070)	промислового (0.0108)	необхідно (0.0058)	проникність (0.0070)	організм (0.0065)
зображення (0.0036)	теорема (0.0054)	оптимальних (0.0067)	капіталізації (0.0052)	підвищення (0.0104)	метаматеріал (0.0035)	діелектричний (0.0065)	відображення (0.0060)
штучних (0.0030)	дирихле (0.0052)	стрижень (0.0053)	грошових (0.0051)	бізнес (0.0097)	недолік (0.0031)	хвильового (0.0051)	твори (0.0055)
класичної (0.0029)	довести (0.0051)	програмних (0.0051)	страхового (0.0048)	інноваційного (0.0093)	вихідного (0.0028)	формули (0.0040)	активації (0.0053)
розпізнавання (0.0026)	фільтрації (0.0050)	функціонал (0.0047)	фінансових (0.0047)	продовжуватися (0.0063)	больцмана (0.0028)	твердих (0.0037)	математичне (0.0044)
інвестиційної (0.0026)	існування (0.0047)	поширення (0.0044)	інноваційної (0.0045)	економіки (0.0062)	магнітне (0.0026)	гурса (0.0036)	більшість (0.0041)
генетичний (0.0025)	максвелла (0.0046)	нелінійної (0.0042)	премії (0.0037)	дана (0.0051)	оптичні (0.0026)	після (0.0036)	інформаційно (0.0038)
відбір (0.0024)	світла (0.0039)	акустичних (0.0040)	життя (0.0037)	текст (0.0046)	зони (0.0024)	ефективної (0.0035)	підтвердження (0.0037)
постановка (0.0024)	бігармонічного (0.0039)	похідних (0.0039)	зобов'язання (0.0034)	банківської (0.0046)	збудження (0.0022)	лікування (0.0034)	ємності (0.0037)
кредитного (0.0022)	кордон (0.0039)	формі (0.0033)	звітності (0.0031)	конкурентоспроможності (0.0045)	взаємодії (0.0022)	тривалості (0.0034)	коригування (0.0035)
гібридний (0.0022)	перетворення (0.0035)	оптимальності (0.0030)	страхові (0.0031)	технологія (0.0045)	магнітних (0.0020)	виживання (0.0034)	наступним (0.0035)
інвестора (0.0021)	явному (0.0035)	обчислити (0.0029)	страхової (0.0029)	масив (0.0039)	активних (0.0020)	медіана (0.0033)	дані (0.0034)
класифікації (0.0020)	напівплощині (0.0033)	заходи (0.0029)	достатності (0.0029)	промисловості (0.0039)	фотон (0.0019)	включення (0.0031)	можливі (0.0033)
нейронні (0.0020)	нейману (0.0028)	вагання (0.0029)	претензія (0.0028)	перспективи (0.0038)	межа (0.0019)	діапазон (0.0031)	дві (0.0032)

Таблиця 3.10 – Ймовірнісний розподіл слів за темами, $t = 12$ при 50 епохах

Topic 1:	Topic 2:	Topic 3:	Topic 4:	Topic 5:	Topic 6:
підбір (0.0096)	мембранних (0.0071)	робота (0.0188)	дохідності (0.0043)	заломлення (0.0117)	інноваційного (0.0036)
відповідному (0.0078)	лапласіан (0.0067)	серія (0.0064)	інвестора (0.0038)	метаматеріал (0.0111)	штучні (0.0029)
паралельних (0.0052)	страховий (0.0050)	середнього (0.0049)	політики (0.0037)	випромінювання (0.0046)	обґрунтування (0.0019)
нестійкість (0.0050)	премія (0.0047)	викид (0.0049)	систематизовані (0.0037)	енергії (0.0042)	пошук (0.0019)
спеціальної (0.0048)	вольтерра (0.0040)	сформованого (0.0045)	страховика (0.0036)	акустичного (0.0041)	економічних (0.0018)
механічних (0.0042)	існування (0.0040)	машин (0.0044)	витрати (0.0034)	електрон (0.0039)	базі (0.0018)
певної (0.0039)	просторовим (0.0040)	диференціальних (0.0044)	дохідність (0.0033)	проходження (0.0038)	помилки (0.0017)
ефективного (0.0039)	регулярного (0.0038)	повздовжньої (0.0043)	організація (0.0032)	щільності (0.0038)	клас (0.0016)
динаміці (0.0037)	стохастичному (0.0038)	збудження (0.0041)	найважливішим (0.0032)	матриця (0.0037)	ліквідність (0.0015)
спектр (0.0034)	нелокальні (0.0034)	магістральних (0.0038)	блек (0.0032)	енергоресурси (0.0036)	ротор (0.0015)
верифікація (0.0033)	навантаженні (0.0034)	складання (0.0037)	довгострокових (0.0032)	графена (0.0036)	статистичне (0.0014)
функціональні (0.0029)	кінцевому (0.0031)	лінеаризованої (0.0036)	росії (0.0031)	спектр (0.0036)	адаптації (0.0014)
страхування (0.0029)	нового (0.0029)	excel (0.0036)	інвесторів (0.0030)	зоні (0.0034)	нелінійних (0.0014)
синаптичних (0.0029)	тимчасових (0.0029)	точне (0.0036)	заборгованості (0.0030)	електромагнітного (0.0030)	парадокс (0.0013)
варіаційний (0.0029)	крайовий (0.0029)	локальної (0.0034)	валюта (0.0028)	звукових (0.0030)	інструментарій (0.0013)
Topic 7:	Topic 8:	Topic 9:	Topic 10:	Topic 11:	Topic 12:
досліджується (0.0105)	технологічним (0.0054)	страховий (0.0215)	ігри (0.0028)	напівплощини (0.0165)	адаптивні (0.0066)
страхування (0.0080)	трубопровід (0.0053)	об'єктний (0.0039)	відлік (0.0027)	лікування (0.0026)	чисельне (0.0045)
страхового (0.0032)	грунт (0.0051)	власного (0.0027)	процедура (0.0027)	банку (0.0021)	розуміння (0.0042)
функціонування (0.0030)	діелектричної (0.0049)	початковий (0.0025)	поведінка (0.0024)	мг (0.0020)	піксель (0.0037)
етапи (0.0025)	металевих (0.0041)	викладатися (0.0022)	математичні (0.0022)	хворих (0.0018)	складний (0.0029)
становлення (0.0023)	кераміки (0.0038)	науковий (0.0021)	ідентифікації (0.0021)	залізи (0.0018)	капіталізація (0.0029)
ейлера (0.0022)	матриці (0.0037)	міський (0.0020)	детально (0.0020)	розкладання (0.0018)	аналогічні (0.0028)
торговельних (0.0019)	якобі (0.0037)	розробка (0.0019)	проектування (0.0019)	кг (0.0018)	вінера (0.0025)
економічний (0.0019)	тангенс (0.0036)	житло (0.0019)	пласт (0.0019)	пацієнтів (0.0017)	стаціонарному (0.0023)
сформулювати (0.0018)	проникність (0.0034)	напрямки (0.0018)	розмірність (0.0018)	виживання (0.0016)	внутрішньому (0.0022)
тариф (0.0018)	інститут (0.0032)	достатності (0.0017)	продовжуватися (0.0017)	мл (0.0016)	нелінійності (0.0022)
парних (0.0016)	концентрація (0.0032)	стратегії (0.0017)	змінний (0.0017)	день (0.0016)	цікавість (0.0022)
генератор (0.0015)	похибки (0.0028)	виготовлення (0.0016)	станція (0.0016)	безризикового (0.0015)	фокусування (0.0022)
установки (0.0015)	дані (0.0027)	налаштування (0.0016)	приділятися (0.0016)	розпізнавання (0.0015)	залучення (0.0022)
дарба (0.0015)	клітина (0.0025)	нейману (0.0015)	необхідність (0.0016)	вибірки (0.0014)	плоском (0.0021)

Після аналізу структури можемо прийти до наступних висновків :

– топік 1 – тема 5) math_phys (математична фізика);

– топік 2 – тема 3) electrodynamic (електродинаміка);

– топік 3 – тема 1) actuar_math (актуарна математика) та тема 7) optimal_control (оптимальне керування);

– топік 4 – тема 2) cluster_analysis (кластерний аналіз);

– топік 5 – тема 6) neural_network (нейронні мережі) та тема 4) investing (інвестування);

– топік 6 – тема 1) actuar_math (актуарна математика);

– топік 7 – тема 4) investing (інвестування).

Як бачимо, деякі з виділених топіків дуже добре відповідають фактичним темам і однозначно їх описують (топік 1, топік 2, топік 4, топік 6, топік 7). Інші топіки виходячи з наборів їх найуживаніших слів або відповідають декільком фактичним темам (топік 3 та топік 5). Виходячи з цього можна зазначити, що при своїх недоліках алгоритм добре впорався з завданням по розподілу тем.

Порівнюючи результати, отримані для таблиць 3.6 – 3.10 можна зробити висновки, що найбільш вдалим виявилось розділення на 7 топіків при 50 епохах навчання (таблиця 3.7). Саме у цьому випадку структура науживаніших слів у топіках найбільш точно описує фактичні тематики документів.

Проаналізуємо, наскільки розподіл документів за топіками відповідає реальним темам документів в датасеті. В таблиці 3.11 наведено ймовірності належності окремих документів досліджуваної колекції до кожного з топіків (при $t = 7$ та кількості епох навчання рівній 50) і фактичні теми цих документів.

Таблиця 3.11 – Ймовірності належності документів колекції окремим топікам

Документ	Ймовірнісний розподіл за топіками	Реальна тема
1	2	3
(27) Приклад дослідження технологічного процесу нагрівання порційної подачі сировинного продукту показати керовані системи досягати мети управління шляхом здійснення технологічних операцій показати керовані системи націлити досягнення єдиної кібернетичної мети доданої вартості запропонувати методика прямого визначення оптимального управління.	(0.975; 0.004; 0.008; 0.000; 0.001; 0.005; 0.006)	5) math_phys
(30) Розглянути задачу оптимального керування тепловим режимом будівлі неробочий час знайти алгоритм оптимального керування режим переривчастого опалення розробити метод побудови конкретних графіків подачі теплота.	(0.991; 0.002; 0.000; 0.001; 0.001; 0.000; 0.005)	5) math_phys
(286) Розробити алгоритмічне програмне забезпечення узагальненого інверсного інтервального методу глобальної умовної оптимізації також метод застосування рішення задачі знаходження оптимального програмного керування нелінійний детермінованими безперервний динамічний система розробити узагальнена модульний схема алгоритм переслідування маневруючої мети перехоплювачем	(0.005; 0.017; 0.004; 0.089; 0.015; 0.86; 0.01)	7) optimal_control
(1204) Стаття присвячена теоретичним питання опису модульних нейронних мереж допомоги спрямованих графів основі запропонованого подання дані визначення цикл провести класифікація розглянути властивості найбільш цікавих практичної точки зору типів цикл запропоноване уявлення призначити загального аналіз.	(0.004; 0.004; 0.006; 0.007; 0.958; 0.017; 0.002)	6) neural_network
(1272) Розглядаються способи попередньої обробки даних різнотипних ознак простір мінімізації конфігурації нейронної мережі розв'язання задача розпізнавання вчитель	(0.001; 0.000; 0.000; 0.002; 0.973; 0.022; 0.001)	6) neural_network
(2201) У статті розглядаються питання прибутковості цінних паперів, що враховують умови їх випуску та положення дивідендної політики емітентів, формування прибутковості диверсифікованого портфеля.	(0.005; 0.005; 0.028; 0.009; 0.009; 0.012; 0.932)	4) investing

Як видно з таблиці 3.11, всі документи, які аналізувались, мають максимальну ймовірність належності до того топіка, ключові слова за яким відповідають його фактичній тематиці. Це свідчить про високу якість роботи обраного методу тематичного моделювання. Дійсно, виходячи з структури ключових слів за топіками у таблиці 3.7, та їх подальшому аналізу в нашій роботі можемо зробити наступні висновки за результатами таблиці 3.11:

– документ (27) з фактичною темою 5) math_phys однозначно відноситься алгоритмом до топіка 1, який відповідає фактичній темі 5) math_phys (математична фізика);

– документ (30) з фактичною темою 5) math_phys однозначно відноситься алгоритмом до топіка 1, який відповідає фактичній темі 5) math_phys (математична фізика);

– документ (286) з фактичною темою 1) actuar_math з найбільшою ймовірністю відноситься алгоритмом до топіка 6, який відповідає фактичній темі 1) actuar_math (актуарна математика);

– документ (1204) з фактичною темою 6) neural_network з найбільшою ймовірністю відноситься алгоритмом до топіка 5, який відповідає фактичним темам 6) neural_network (нейронні мережі) та 4) investing (інвестування);

– документ (1272) з фактичною темою 6) neural_network з найбільшою ймовірністю відноситься алгоритмом до топіка 5, який відповідає фактичним темам 6) neural_network (нейронні мережі) та 4) investing (інвестування);

– документ (2201) з фактичною темою 4) investing однозначно відноситься алгоритмом до топіка 7, який відповідає фактичній темі 4) investing (інвестування).

Виходячи з отриманих результатів, можемо сказати, що реалізована модель добре справляється з задачею визначення ймовірнісних розподілів наукових документів, поданих своїми анотаціями, за темами.

Висновки за розділом 4

Було проведено ряд досліджень з тематичного моделювання для заданого датасета для 6, 7, 8 та 12 топіків. Експерименти показали, що найкраще модель відпрацьовує, коли обрана кількість топіків співпадає з кількістю наявних тем. Проаналізувавши розподіл датасету на 7 топіків на 50 та 300 епохах навчання, прийшли до висновку, що на 50 епохах програма видавала кращі результати. Це можливо обумовити перенавчанням програми, після котрого вона починає робити більше помилок.

Також було визначено значення похибки для кожного з модулів (Generator, Discriminator, Encoder) для розподілу датасету на 6, 7, 8 та 12 топіків. Аналіз значень похибок дозволив підтвердити, що при обранні кількості топіків, яка відповідає реальній кількості тем (7 топіків), за 50 епох навчання програма видавала найкращі результати.

Було пораховано ймовірності належності окремих документів досліджуваної колекції до кожного з топіків для випадку розподілу на 7 топіків за 50 епох навчання. Результати аналізу обраних з датасету документів підтвердили високу якість обраного методу розподілу.

ВИСНОВКИ

Під час виконання даної кваліфікаційної роботи була проаналізована проблема тематичного моделювання текстових документів та був проведений системний аналіз цієї проблеми. Застосовуючи метод аналізу ієрархій, було визначено оптимальний метод розв’язання задачі тематичного моделювання наукових текстів. Вибір вказаної моделі обумовлено її широким застосуванням та популярністю у схожих задачах.

Отримані під час проведення системного аналізу результати демонструють, що найбільш ефективним за певними критеріями методом оцінки тематичної структури колекції документів є нейромережевий підхід.

У процесі розв’язання визначеної задачі був сформульований алгоритм розподілу за темами, а на його основі використовуючи мову програмування Python 3 розроблено програмний продукт на основі моделі BAT. Це дозволило ефективно впровадити процес визначення тем і ключових слів для заданої колекції документів.

Розроблений алгоритм показав прийнятні результати щодо розподілу тем, виявився зручним у використанні для проведення досліджень.

Важливо зауважити, що у кваліфікаційній роботі було досліджено розв’язання задачі тематичного моделювання для текстових документів, написані українською мовою, котра входить до східнослов'янської групи мов. В подальшому можна досліджувати можливість застосування даного підходу для інших груп мов.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. FitzGerald J., FitzGerald A. F. Fundamentals of Systems Analysis: Using Structured Analysis and Design Techniques. London : Wiley, 1987. 899 p.

2. Сорока К. О. Основи теорії систем і системного аналізу. Харків, "ХНАМГ", 2004. 115 с.

3. Катренко А. В., Пасічник В. В., Пасько В. П. Теорія прийняття рішень. Київ : Видавнича група BHV, 2009. 448 с.

4. Катренко А. В. Системний аналіз. Львів : “Новий світ – 2000”, 2011. 396 с.

5. Topic Modeling for The New York Times News Dataset. URL: https://towardsdatascience.com/topic-modeling-for-the-new-york-times-news-dataset-1f643e15caac (дата звернення: 26.05.2022).

6. Knowledge discovery through directed probabilistic topic models: a survey / A. Daud, J. Li, L. Zhou, F. Muhammad. *Frontiers of Computer Science in China*. 2010. V. 4, № 2. P. 280–301.

7. Topic Modeling with LSA, PLSA, LDA & lda2Vec. URL: https://medium.com/nanonets/topic-modeling-with-lsa-psla-lda-and-lda2vec-555ff65b0b05 (дата звернення: 26.05.2022).

8. Lerch F., Ultsch A., Lötsch J. Distribution Optimization: An evolutionary algorithm to separate Gaussian mixtures. *Distribution Optimization*. 2020. P. 1–15.

9. Wang R., Hu X., Zhou D., He Y., Xiong Y., Ye C., Xu H. Neural Topic Modeling with Bidirectional Adversarial Training. 2020.

10. Neural_Topic_Models. URL: https://github.com/zll17/Neural_Topic_Models (дата звернення: 04.01.2024)

11. Стецун К. С. Розв’язання задачі тематичного моделювання шляхом розділення сумішей ймовірнісних розподілів. Матеріали 26-го міжнародного молодіжного форуму *«Радіоелектроніка і молодь у XXI столітті»* (20 грудня 2022). С. 45-46.

12. Стецун К., Гибкіна Н., Шпакович М. Розв’язування задачі тематичного моделювання наукових текстів шляхом розділення сумішей

ймовірнісних розподілів. Матеріали статей Міжнародної науково-практичної конференції *«Інформаційні технології та комп'ютерне моделювання»* (15-16 грудня 2022). С. 74-76.

13. Стецун К. Розв'язання задачі кластеризації наукових текстів. Матеріали 27-го міжнародного молодіжного форуму *«Радіоелектроніка і молодь у XXI столітті»* (10-12 травня 2023). С. 201-202.

14. Stetsun K. Topic detection and analysis in scientific texts as a problem of separating probability distribution mixtures. Матеріали другої міжнародного конференції *«LEARNING & TEACHING: after War and during Peace»* (10 листопада 2023).