НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

"На правах рукопису"
УДК 681.513.7

До захисту допущено
Завідувач кафедри

_____ Олександр РОЛІК

06 “червня” 2022 р.

МАГІСТЕРСЬКА ДИСЕРТАЦІЯ

на здобуття ступеня магістра

за освітньо-науковою програмою

«*Інформаційні управляючі системи та технології*»

зі спеціальності 126 «*Інформаційні системи та технології*»

на тему:

«Інформаційна система аналізу змісту новин та прогнозування подій на його основі»

Виконав:
студент ІІ курсу, групи **ІС-01мн**
Процюк Юрій Володимирович _____

Керівник:
доцент, к.ф.-м.н, доцент,
Гавриленко Олена Валеріївна _____

Консультант:
_____

Рецензент:
доцент кафедри ІПІ, к.т.н.,
Олійник Юрій Олександрович _____

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.

Студент _____

Київ – 2022 року

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

Рівень вищої освіти – другий (магістерський)

Спеціальність – 126 «*Інформаційні системи та технології*»

Освітньо-наукова програма *«Інформаційні управляючі системи та технології»*

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Олександр РОЛІК

«06» червня 2022 р.

ЗАВДАННЯ
на магістерську дисертацію студенту

Процюка Юрія Володимировича

1. Тема дисертації «Інформаційна система аналізу змісту новин та прогнозування подій на його основі», науковий керівник дисертації Гавриленко Олена Валеріївна, к.ф.-м.н., доцент, затверджені наказом по університету від «26» квітня 2022 р. № НС/88/2022

2. Термін подання студентом дисертації 08.06.2022

3. Об'єкт дослідження – зміст новин.

4. Предмет дослідження – інформаційна система аналізу змісту новин та прогнозування подій.

5. Перелік завдань, які потрібно розробити: аналіз та узагальнення наукових досліджень за темою роботи, аналіз наявних підходів розв'язання задачі, формулювання постановки задачі, дослідження проблеми якісних новин, розробка методів аналізу новин та прогнозування подій на його основі, розробка інформаційної системи, проведення аналізу результатів дослідження.

6. Орієнтовний перелік графічного (ілюстративного) матеріалу: схема структурна класів програмного забезпечення; схема структурна компонентів інфомаційної системи; схема критеріїв якості ЗМІ; графік визначення кількості кластерів; графіки побудованих кластерів; матриці подібності кластерів; хмари слів для кластерів; побудовані ланцюги асоціативних правил.

7. Орієнтовний перелік публікацій

Стаття у міжвідомчому науково-технічному збірнику «Адаптивні системи автоматичного управління» 1 (40) та публікація тез у збірнику наукової конференції студентів науково-практичної конференції молодих вчених та студентів «Інженерія програмного забезпечення і передові інформаційні технології» – SoftTech-2021.

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

9. Дата видачі завдання «31» січня 2022 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Аналіз та узагальнення наукових досліджень за темою дисертації	14.02.2022	
2	Аналіз наявних підходів розв’язання задачі	25.02.2022	
3	Формулювання постановки задачі	10.04.2022	
4	Розробка методів аналізу новин та прогнозування подій на його основі	17.04.2022	
5	Розробка інформаційної системи	06.05.2022	
6	Проведення аналізу результатів дослідження	13.05.2022	
7	Оформлення пояснювальної записки	20.05.2022	
8	Подання роботи на попередній захист	25.05.2022	
9	Подання роботи на основний захист	13.06.2022	

Студент Юрій ПРОЦЮК

Науковий керівник Олена ГАВРИЛЕНКО

РЕФЕРАТ

Магістерська дисертація: 107 сторінок, 4 розділи, 49 рисунків, 4 таблиці, 4 додатки, 43 джерела.

Актуальність обумовлюється збільшенням кількості продукованої інформації у світі. Значна частина цієї інформації є новинами, що дають певну картину світу і мають вплив на уми людей. Змога знати, що відбуватиметься в майбутньому, дозволить можливо змінювати хід подій, зокрема уникати катастроф. Доречною буде розробка засобу, що дозволятиме прогнозувати подій, що відбуватимуться в майбутньому.

Мета дослідження – підвищення релевантності прогнозів виникнення подій на основі аналізу змісту новин.

Об’єкт дослідження – зміст новин.

Предмет дослідження – інформаційна система аналізу змісту новин та прогнозування подій.

Методи дослідження – методи кластеризації та методи машинного навчання засновані на асоціативних правилах.

Наукова новизна отриманих результатів полягає в розробці та модифікації підходів до прогнозування подій на основі новин, зокрема модифікації асоціативних правил щодо об’єднання їх в ланцюги, які дозволили б виявляти причинно-наслідкові зв’язки в тексті новин.

Публікації. Результати досліджень буде опубліковано в міжвідомчому науково-технічному збірнику «Адаптивні Системи Автоматичного Управління» (на стадії друку) [1] та в тезах наукової конференції студентів науково-практичної конференції молодих вчених та студентів «Інженерія програмного забезпечення і передові інформаційні технології» – SoftTech-2021 [2].

ПРОГНОЗУВАННЯ ПОДІЙ, АНАЛІЗ НОВИН, ЯКІСТЬ НОВИН, КЛАСТЕРИЗАЦІЯ, АСОЦІАТИВНІ ПРАВИЛА, ОБРОБКА ПРИРОДНОЇ МОВИ

ABSTRACTS

Master's dissertation: 107 pages, 4 chapters, 49 figures, 4 tables, 4 appendices, 43 sources.

Topicality is due to the increasing amount of information produced in the world. Much of this information is news that gives a picture of the world and has an impact on people's minds. Being able to know what will happen in the future will make it possible to change the course of events, in particular to avoid catastrophes. It will be appropriate to develop a tool that will allow you to predict future events.

The purpose of the study is to increase the relevance of forecasts of events based on the analysis of news content.

The object of research is the content of news.

The subject of research is the information system of news content analysis and event forecasting.

Research methods – clustering methods and machine learning methods based on associative rules.

The scientific novelty of the obtained results is the development and modification of approaches to forecasting events based on news, in particular the modification of associative rules for combining them into chains, which would reveal the causal links in the news text.

Publications. The research results will be published in the interdepartmental scientific and technical collection "Adaptive Automatic Control Systems" (at the printing stage) [1] and in the abstracts of the scientific conference of students of the scientific-practical conference of young scientists and students "Software Engineering and Advanced Information Technology" - SoftTech -2021 [2].

EVENT FORECASTING, NEWS ANALYSIS, NEWS QUALITY, CLUSTERIZATION, ASSOCIATIVE RULES, NATURAL LANGUAGE PROCESSING

ЗМІСТ

ВСТУП

Засоби масової інформації вже давно перестали бути просто джерелом новин щодо подій у світі. Все частіше можна помітити, як новини передбачають ті чи інші події, ніби накликають їх. Тому постає питання побудови моделі, що допомагатиме прогнозувати подій в майбутньому базуючись на інформації про минуле. Виявлення знань з текстів природними мовами є одним з найважливіших питань інтелектуального аналізу даних [2].

Актуальність обумовлюється збільшенням кількості продукованої інформації у світі. Значна частина цієї інформації є новинами. Найпопулярніші у світі сайти новин відвідують понад 300 мільйонів разів за місяць. Новини перестали бути просто джерелом інформації про те, що сталося. Вони все частіше дають нам певну картину світу і мають суттєвий вплив на уми людей. Можна помітити як новини стають передвісниками подій, що згодом відбуваються в майбутньому. Тобто на основі минулого досвіду, можна визначити, що буде потім. Саме змога знати, що відбуватиметься в майбутньому, дозволить коригувати поведінку і так можливо змінювати хід подій, а саме уникати катастроф в різних сферах: соціальній, економічній, сфері охорони природи та інших.

Тому доречною буде розробка засобу, що дозволятиме прогнозувати подій, що відбуватимуться в майбутньому.

Мета дослідження – підвищення релевантності прогнозів виникнення подій на основі аналізу змісту новин.

Для досягнення мети необхідно виконати такі завдання:

- виконати огляд досліджень та їх результатів для задач за схожою тематикою;
- виконати аналіз проблеми якості ЗМІ;

– представити текстову інформацію новин у форматі, що підходить для подальшої обробки;
– розробити алгоритм попередньої обробки текстових даних;
– розробити алгоритм аналізу вмісту новин;
– розробити алгоритм прогнозування подій на основі новин;
– створити інформаційну систему з аналізу новин та прогнозування подій на їх основі;

Об’єкт дослідження – зміст новин.

Предмет дослідження – інформаційна система аналізу змісту новин та прогнозування подій.

Методи дослідження – методи кластеризації та методи машинного навчання засновані на асоціативних правилах.

Наукова новизна отриманих результатів полягає в розробці та модифікації підходів до прогнозування подій на основі новин, зокрема застосуванні та модифікації алгоритмів кластеризації та алгоритмів побудови асоціативних правил для подальшого об’єднання їх в ланцюги, які дозволили б виявляти причинно-наслідкові зв’язки в тексті новин.

Публікації. Результати досліджень буде опубліковано в міжвідомчому науково-технічному збірнику «Адаптивні системи автоматичного управління» (підтверджено прийом до публікації) [1]. Також результати опубліковані в тезах наукової конференції студентів науково-практичної конференції молодих вчених та студентів «Інженерія програмного забезпечення і передові інформаційні технології» – SoftTech-2021 [2].

1 ОГЛЯД ПІДХОДІВ ТА МЕТОДІВ РОЗВ'ЯЗАННЯ ЗАДАЧІ АНАЛІЗУ НОВИН ТА ПРОГНОЗУВНАННЯ ПОДІЙ НА ЙОГО ОСНОВІ

1.1 Загальна схема розв'язання задачі аналізу новин та прогнозування подій на їх основі

Задача аналізу новин та прогнозування подій на їх основі комплексна, але одначе можна виокремити такі етапи роботи (рисунок 1.1):

а) попередня обробка тексту новини;

б) визначення ключових слів;

в) кластеризація чи класифікація новин за тематикою;

г) оцінка виконаної класифікації/кластеризації;

д) корегування параметрів алгоритму;

е) аналіз новин і побудова прогнозу;

ж) перевірка якості прогнозування;

з) модифікація алгоритму прогнозування.

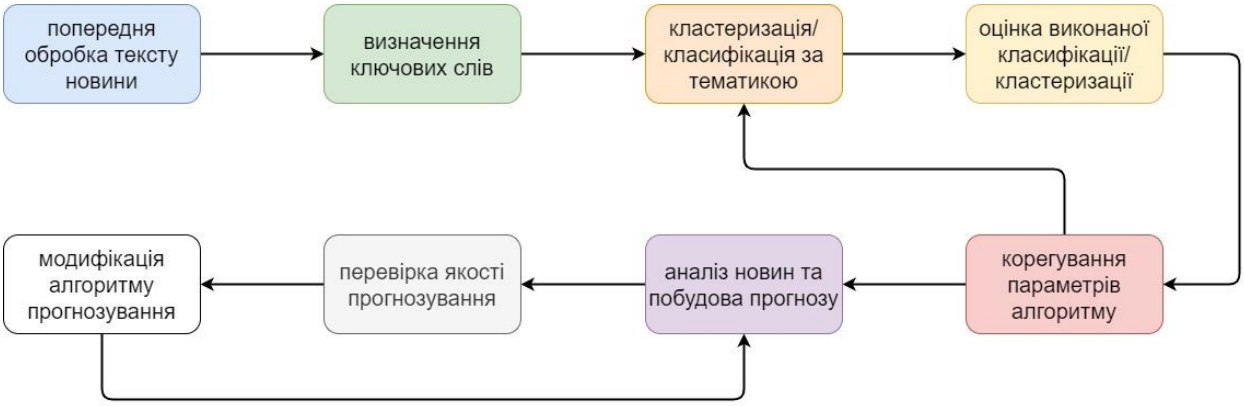

Рисунок 1.1 – Загальна схема розв'язання задачі

1.2 Огляд підходів попередньої обробки текстових даних

Попередня обробка тексту означає просто привести його в форму, що буде зрозуміла і передбачувана для задачі. Ідеальна попередня обробка для однієї задачі

може стати жахіттям для іншої. Саме з цієї причини попередню обробку не можна переносити від однієї задачі до іншої напряму. Наприклад, при пошуку часто вживаних слів в новинах не варто видаляти стоп-слова, що може бути корисним при пошуку ключових слів чи задачі визначення змісту [3].

Одним з базових методів попередньої обробки є перетворення всього тексту до нижнього регістру. Цей метод найпростіший і дуже ефективний. Він може застосовуватися до більшості задач інтелектуального аналізу тексту та обробки природної мови, може допомогти, коли набір даних не дуже великий, особливо він забезпечує узгодженість очікуваного виходу. Якщо набір даних невеликий і різні варіанти написання слова дають різні виходи, саме тоді переведення до нижнього регістру стуттєво допомагає справитися з проблемою розрідженості даних. Проте і практика переведення до нижнього регістру не є універсальною, бо іноді одне слово написане з великої літери й слово написане з малої мають абсолютно різний семантичний зміст, до прикладу слово “System” в мові програмування Java суттєво відрізняється від слова “system” в мові Python.

Стемінг – важливий метод, що полягає в скороченні словоформи до кореня. Причому для попередньої обробки тексту корінь – не обов’язково морфема, а може бути просто канонічна найчастіше вживана форма вихідного слова. Стемінг – суто евристичний процес, що відкидає кінці слів, причому його мета – правильно перетворити слово в кореневу форму. Існує чимало алгоритмів стемінгу. Найбільш поширеним, який водночас є ефективним для англійської мови, що доведено багатьма експериментами, є алгоритм Портера. Стемінг корисний для розв’язання проблеми розрідженості та для стандартизації словника.

Лематизація – процес дуже близький до стемінгу, який видаляє закінчення слова і зіставляє його з кореневою форму, але часто насправді просто обрубує слова. Проте лематизація – дещо складніша, вона передбачає правильне визначення кореневої форми, тобто для дієслова це буде інфінітив, а для іменника – однина.

Лематизація може використовувати словники, як, наприклад, WordNet, для зіставлення чи деякі особливі підходи, заснованих на правилах. На практиці лематизація не завжди дає очікувану перевагу для задачі пошуку чи класифікації тексту, тому слід зважати на швидкість роботи алгоритмів, адже в багатьох випадках підійде і звичайний стемінг, але варто спробувати обидва підходи, щоб перевірити чи суттєво покращує лематизація метрику продуктивності. На рисунку 1.2 показано різницю між стемінгом та лематизацією.

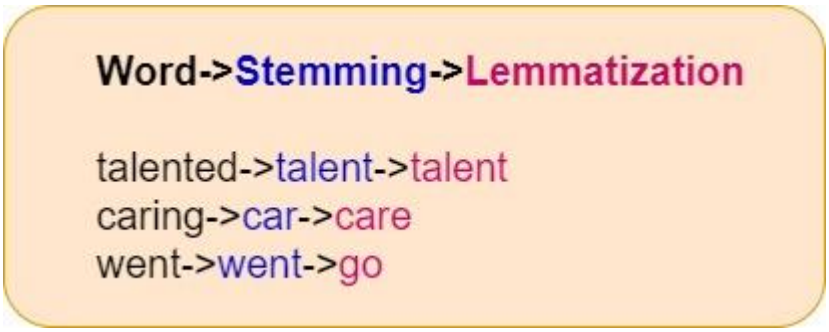

Рисунок 1.2 – Приклад стемінгу і лематизації слова

Мета видалення стоп-слів полягає в тому, щоб зосередитися на важливих словах тексту, що краще допоможе зрозуміти його зміст. Дослідження вказують на те, що для задач класифікації видалення стоп-слів не відіграє суттєвої ролі, але може допомогти хоча б зменшити розмір тексту, а як наслідок і моделі. Для видалення стоп-слів можна використовувати готові набори таких слів чи самостійно створити список стоп слів [4].

Методом попередньої обробки тексту, що дозволяє дещо покращити результати класифікації є нормалізація тексту. Це процес, що перетворює текст в стандартну канонічну форму. Це може бути, як і виправлення орфографічних помилок, заміна жаргонізмів чи діалектизмів літературними словами, так і уніфікації різних форм написання одного і того ж слова. Недоліком нормалізації є відсутність стандартного способу, на відміну від тих же стемінгу та лематизації. Однак загальні підходи до нормалізації тексту передбачають зіставлення словників,

статистичний машинний переклад і підходи засновані на виправленні орфографії. Це може бути корисним при роботі з текстами, де часто зустрічаються синоніми, нормалізація такого тексту допоможе в його подальшому аналізі.

Максимально простим і водночас обов’язковим є вилучення шуму з тексту, тобто видалення спеціальних символів, цифр. Як правило, вони не несуть змістового навантаження і навпаки можуть завадити правильному розумінню інших слів.

Варто згадати й один з найскладніших підходів – збагачення (збільшення) тексту. Воно передбачає доповнення текстових даних інформацією, якої раніше не було і може надати більшої семантики тексту, покращивши глибину його аналізу та прогностичні можливості. Цей підхід актуальний, наприклад, для формування пошукових запитів, але може бути шкідливим для аналізу текстових даних з метою виявлення закономірностей [5].

На рисунку 1.3 показано залежність між складністю та обов’язковість застосування того чи іншого методу попередньої обробки тексту. Методи, що використовуватимуться виділені зеленим кольором.

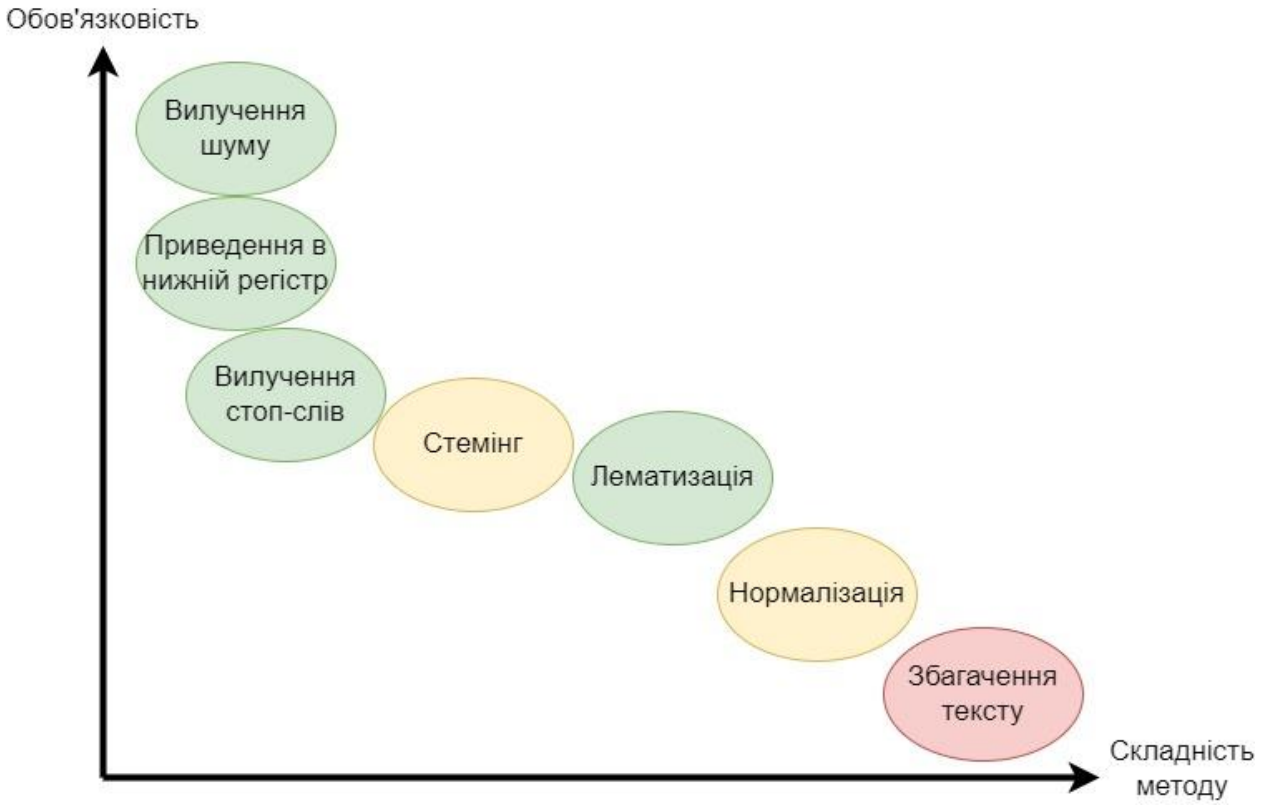

Рисунок 1.3 – Графік обраних методів попередньої обробки тексту

Окремою складовою попередньої обробки тексту є токенізація, що передбачає сегментацію тексту на дрібніші шматки: слова, словосполучення, речення. Токенізація не є універсальною процедурою у зв’язку з тим, що спецсимволи можуть нести семантичне навантаження чи, наприклад, можуть існувати різні способи розбиття тексту на словосполучення.

1.3 Підходи кластеризації текстової інформації

Кластеризація – це процес класифікації об’єктів на підмножини, що мають значення в контексті певної проблеми. Тому об’єкти об’єднуються в ефективне представлення, яке і характеризує вибрану сукупність. Відношення між об’єктами представлені у вигляді матриці близькості, в якій рядки й стовпці відповідають об’єктам. Якщо об’єкти характеризуються як точки в d-вимірному метричному просторі, тоді близькість може бути визначена через пошук евклідових відстаней.

Якщо не встановлена значуща міра відстані чи близькості між об’єктами, то значущий кластерний аналіз неможливий. Матриця близькості – єдиний вхід для алгоритмів кластеризації. Як вже було зазначено, кластеризація – особливий вид класифікації. Ключовою відмінністю кластеризації від класифікації, що і є причиною використання першої в даній роботі, є відсутність наперед наявних категорій, на які потрібно поділити дані.

Кластеризація тексту – задача групування текстів без міток так, щоб тексти в одному наборі були схожі один на одного більше, ніж на тексти інших наборів (кластерів). Алгоритми кластеризації обробляють тексти й визначають, чи існують в наборі даних природні групи (кластери).

Представлені у векторному вигляді тексти порівнюють між собою, вимірюючи відстань між векторами ознак. Вектори, що знаходяться поруч, повинні належати один одному, якщо ж відстань між векторами велике – вони мають знаходитися в різних кластерах. Кластеризація охоплює три аспекти, як і зображено на рисунку 1.4. Цільова функція повідомляє, що існують найкращі кластери та призупиняє подальшу обробку.

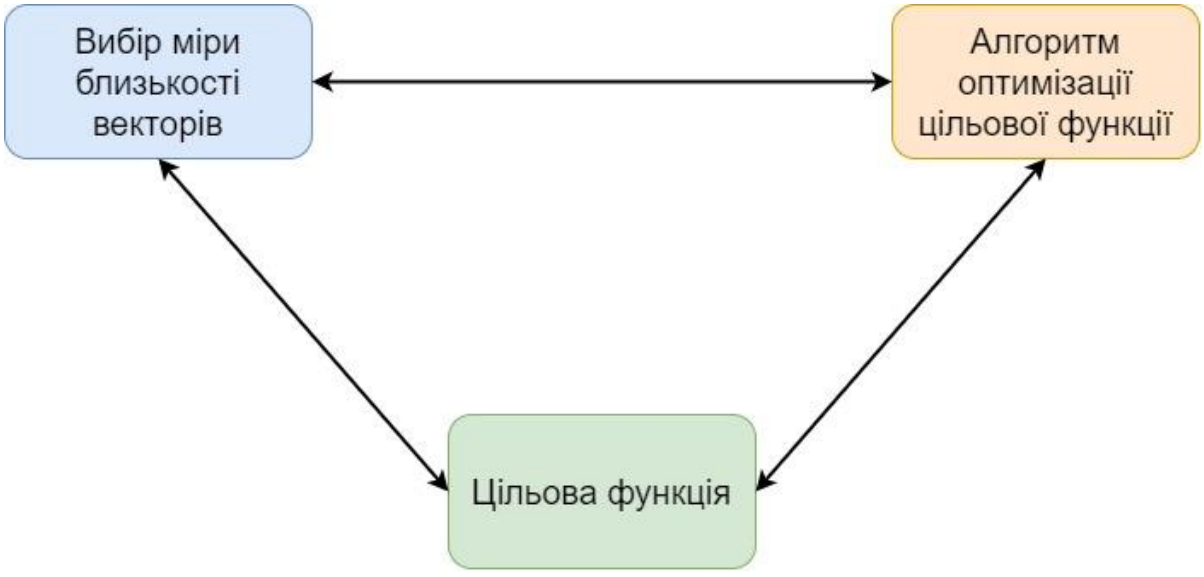

Рисунок 1.4 – Основні аспекти кластеризації

Загалом кластеризацію можна розділити на дві великих групи. Жорстка кластеризація групує елементи так, щоби кожен елемент належав лише одному кластеру. Метод к–середніх – алгоритм жорсткої кластеризації даних. Своєю чергою, м’яка кластеризація дозволяє, щоб елемент належав більше ніж одній групі (кластеру). Прикладом м’якої кластеризації є нечітка кластеризація.

Кластеризація тексту може відбуватися на рівні документів, речень чи навіть слів. Кластеризація на рівні документів відповідає вимогам цієї роботи, адже служить для групування документів за однією темою.

Перетворення текстових даних у вектори з реальними значеннями називається вилученням ознак. Найпростішою технікою представлення тексту є мішок слів. У нього складається список унікальних слів в текстах. Тоді кожен документ можна представити у вигляді вектора, в якому якщо слово наявне стоїть 1 на відповідному місці й 0 в разі відсутності слова. Можна також підрахувати скільки разів кожне слово зустрічається в тексті. Популярним підходом є використання методу визначення частоти й зворотної частоти документу (TF-IDF). Поширеним методом векторного представлення тексту є модель word2vec.

Подібність при кластеризації тексту можна рахувати базуючись на лексичній чи семантичній подібності слів. Слова подібні лексично, якщо в них схожа послідовність символів. Часто слова з різним значенням схожі за написанням, що і робить цей підхід недосконалим. Між словами існує семантична подібність, якщо вони мають схожі чи близькі значення. В цьому випадку потрібно користуватися алгоритмами на основі корпусів даних [6].

Існує велика кількість алгоритмів кластеризації тексту, навіть якщо не розглядати використання нейронних мереж (рисунок 1.5). Перш за все, це ієрархічний – при розбитті на кластери починаємо з одного великого, який ділимо на підкластери. Прикладами є DIANA і MONA. Протилежним ієрархічному є агломеративний підхід, який спершу розглядає кожен текстовий документ як

окремий кластер, а потім об’єднує схожі в більші кластери. Алгоритми – BIRCH і CURE. Кількість кластерів задається в алгоритмах розбиття, прикладами є метод к–середніх, ISODATA, PAM. Алгоритми на основі щільності формують кластери на основі того, скільки точок попадає в заданий радіус (DBSCAN) [7]. При ймовірнісному підході група слів належить темі й задача полягає в тому, щоб ідентифікувати ці теми. Слова також мають ймовірність належності темі. Тематичне моделювання окрема задача обробки природних мов, яка все-таки схожа на м’яку кластеризацію. Для цієї задачі використовують ймовірнісний латентно-семантичний аналіз та латентне розміщення Діріхле.

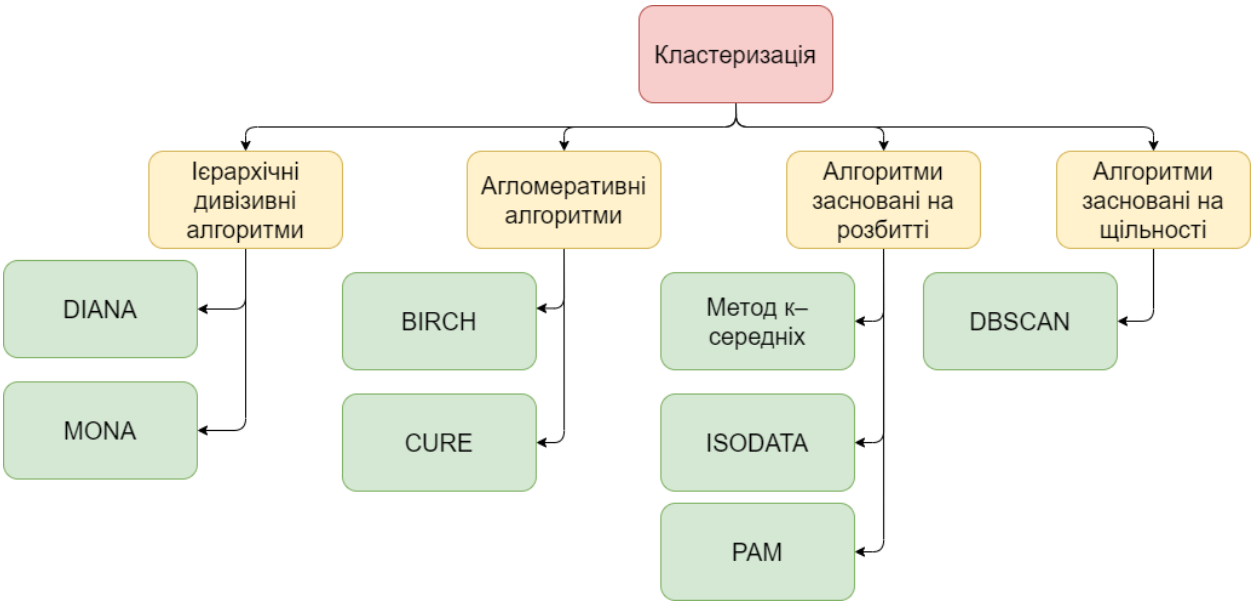

Рисунок 1.5 – Алгоритми кластеризації

Не менш важливою за сам алгоритм кластеризації є оцінка його якості, яку можна оцінити двома шляхами. Міра зовнішньої якості потребує опитування користувачів програмного продукту тому не підходить на етапі первісної побудови моделі. Міра внутрішньої якості оцінює кластеризацію на основі результатів роботи алгоритму, тобто порівнює структуру кластерів і їх відношення один до одного. Компактність вимірює як близько точки згруповані в кластері, а розподіл –

наскільки кластери відрізняються між собою. Тобто компактність – внутрішньокластерна дисперсія, а розподіл – відстань між кластерами [8].

1.4 Огляд методів аналізу вмісту новин та прогнозування

Оскільки задача аналізу вмісту новин та прогнозування подій комплексна, не існує єдиного підходу до її розв'язання. Варто виокремити такі методи аналізу тексту та його подальшої обробки (рисунок 1.6):

- спеціально написані регулярні вирази;
- класифікатори:
  - генеративний: наївний баєсівський класифікатор;
  - дискримінативний: моделі максимальної ентропії;
- послідовні моделі:
  - рекурентні нейронні мережі;
  - прихована марковська модель;
  - умовна марковська модель;
  - умовні випадкові поля;
- інші підходи засновані на машинному навчанні:
  - асоціативні правила.

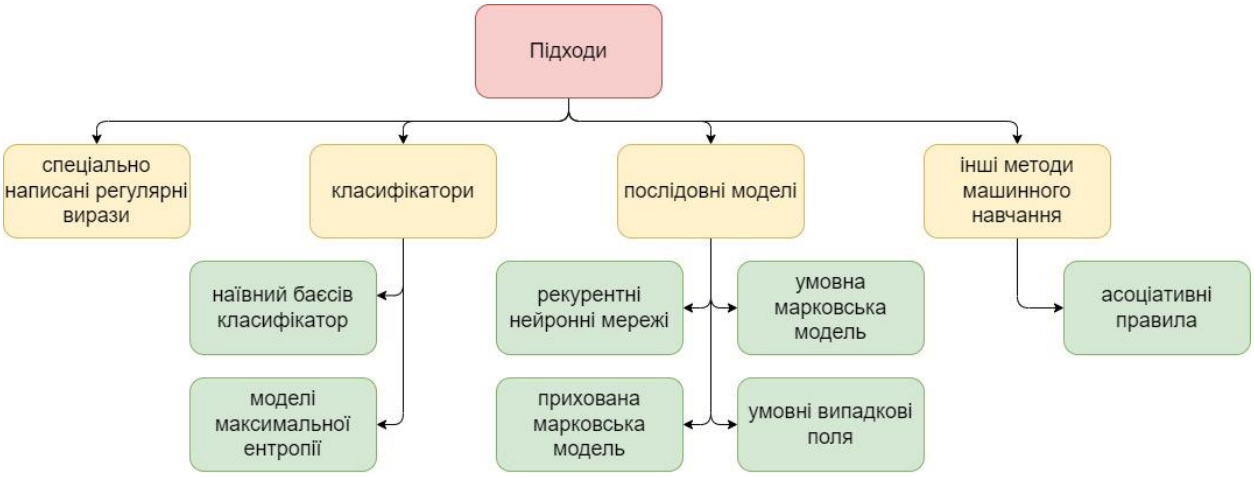

Рисунок 1.6 – Підходи до аналізу тексту та прогнозування на його основі

Регулярні вирази – найпростіший метод, що на основі певних синтаксичних шаблонів допомагає виконувати пошук по тексту. Не підходить для використання у задачі аналізу новин та прогнозування на їх основі через неможливість задання всіх наявних шаблонів.

Наївний баєсів класифікатор – простий імовірнісний класифікатор, що допомагає віднести текст до якогось із заданих класів, використовуючи теорему Баєса з припущенням незалежного виникнення подій (слів у тексті). Знову ж таки не підходить для задачі, що розв'язується в даній роботі, через те, що показує гарні результати класифікації на лише невеликих наборах простих даних.

Поліноміальна логістична регресія корисна для випадків, коли потрібно класифікувати об'єкти на основі значень набору змінних, у випадку тексту його векторного представлення. Підхід подібний простій логістичній регресії, але більш загальний, оскільки залежна змінна, клас тексту, не обмежена двома змінними. Однак, не підходить для розв'язання поставленої задачі через те, що неможливо наперед задати класи текстів.

Нейронні мережі – поширений метод машинного навчання, який вирішує абсолютно різні типи задач, в тому числі пов'язаних з аналізом текстів. Власне для цього використовуються рекурентні нейронні мережі, у якому вузли з'єднані в орієнтований у часі граф. Це дозволяє мережі проявляти динамічну поведінку, що і використовується при роботі з текстовою інформацією.

Прихована та умовна марковські моделі – це статистичні послідовні моделі. Тобто враховуючи послідовність вхідних даних, слів, марковські моделі обрахує послідовність даних однакової довжини. Марковська модель – це граф, вузли якого надають мітки, а ребра вказують на ймовірність переходу від одного вузла до іншого. Разом вони використовуються для підрахунку послідовності міток з врахуванням вхідної послідовності. Марковські процеси в задачі обробки тексту найчастіше використовуються при маркуванні частин мови та розпізнаванні

іменованих сутностей, де мітка залежить не лише від слова, що й розглядається, а й послідовності міток до цього часу [9].

Умовні випадкові поля – це частковий випадок марковського випадкового поля. Це дискримінаційна модель, яка підходить для задач прогнозування, коли контекстна інформація або відстань сусідів впливають на прогноз. Застосовуються в розпізнаванні іменованих сутностей, маркуванні частин мови, прогнозуванні генів та інших задачах обробки текстів.

Проте найкраще для розв'язання задачі аналізу новин та прогнозування подій на їх основі підходять асоціативні правила. Це оператори вигляду «якщо-то», що допомагають показати ймовірність взаємозв'язку між елементами даних у великих наборах даних в різноманітних типах сховищ і баз даних. Інтелектуальний аналіз асоціативних правил має значну кількість сфер застосування і ним послуговуються для виявлення кореляції продаж в транзакційних даних чи наборах медичних даних.

В науці про дані асоціативні правила використовуються для пояснення здавалось би таких незалежних сховищ інформація, як-от реляційні та транзакційні бази даних. Саме використання асоціативних правил часом називають «інтелектуальним аналізом асоціативних правил». У медицині лікарі можуть користуватися асоціативними правилами з метою діагностики пацієнтів. При постановці діагнозу слід враховувати безліч факторів-змінних, позаяк багато захворювань мають спільні чи схожі симптоми. Використовуючи асоціативні правила й аналіз даних на основі машинного навчання, лікарі можуть визначити умовну ймовірність певної хвороби, для цього вони порівнюють взаємозв'язок симптомів з даних про минулі випадки. При постановці нових діагнозів розроблена модель може адаптувати правила для врахування змін.

Дизайн взаємодії з користувачем – розробники можуть збирати дані про те, як відвідувачі використовують певний сайт. Потім можна скористатися асоціативними правилами для оптимізації користувацького інтерфейсу вебсайту чи

мобільного застосунку, провівши аналіз, де користувачі найчастіше натискають і що підвищує ймовірність того, що вони куплять товар чи вчинять певну дію.

У торгівлі можна збирати даних про моделі покупок, фіксуючи дані після того, як штрих-коди скануються на касах. Моделі машинного навчання шукатимуть збіг даних, щоб визначити, які продукти скоріш за все будуть придбані разом і з якою ймовірністю. На основі цих даних, продавці зможуть скоректувати стратегію маркетингу і продажів. Проте чи не найяскравішим є приклад використання асоціативних правил в індустрії розваг. Сервіси на зразок Netflix чи Spotify можуть використовувати асоціативні правила для підтримки своїх механізмів рекомендацій. Моделі машинного навчання аналізують історичні дані поведінки користувачів щодо прихованих закономірностей, розробляють асоціативні правила і використовують їх, щоб рекомендувати контент з яким користувач, ймовірно, взаємодіятиме. Звісно, що можна організувати контент так, щоб першим показувати найцікавіший для користувача.

На базовому рівні інтелектуальний аналіз асоціативних правил включає використання моделей машинного навчання для аналізу даних на предмет шаблонів чи збігів в базі даних. Він визначає поширені асоціації «якщо-то», які і є асоціативними правилами. Асоціативне правило складається з двох частин: антецедент, (іншими словами причина, попередник, якщо), і консеквент (наслідок, то). Антецедент – це елемент знайдений в даних. Консеквент – елемент, який знайдений в поєднанні з антецедентом.

Правила асоціації створюються шляхом пошуку в даних частих причинно-наслідкових зв’язків «якщо-то» і використання критерії затребуваності й впевненості для визначення найважливіших взаємозв’язків. Затребуваність – це показник того, як часто правило з’являється в даних. Впевненість показує, чи і як часто твердження «якщо-то» виявилися справедливими. Третій показник ліфт, який використовується для порівняння впевненості з очікуваною достовірністю або того,

скільки разів «якщо-то» буде визначене істинним. Асоціативні правила розраховуються на наборі елементів мінімального розміру 2. Якщо правила будуються на основі аналізу всіх можливих наборів елементів, правил може бути забагато, що вони не матимуть великого значення. При цьому асоціативні правила найчастіше створюються з правил, які добре представлені у вхідному наборі даних.

Як уже було сказано, сила правила асоціацій визначається двома основними параметрами – затребуваністю і впевненістю. Правило може показати сильну кореляцію в наборі тестових даних, але на практиці зустрічатися рідко. Це означає високу затребуваність, проте низьку впевненість. Навпаки, правило в тестовому наборі може бути рідкісним, але на практиці зустрічатися часто, це свідчить про високу впевненість, але низьку затребуваність. Використання цих показників допомагає аналітикам відділити причинно-наслідкові зв'язки від кореляції та дозволяє оцінити це правило. Ліфт, що є третім параметром, – це відношення впевненості до затребуваності. Якщо значення ліфту від'ємне, то між точками даних є від'ємна кореляція, якщо значення додатне – то існує позитивна кореляція. Лише коли коефіцієнт рівний нулю, то кореляції нема.

Серед популярних алгоритмів, що використовують асоціативні правила є AIS, SETM, Apriori та їх варіації.

За допомогою алгоритму AIS набори елементів створюються і підраховуються при скануванні даних. В транзакційних даних алгоритм AIS визначає, які великі набори елементів містять транзакцію, а нові набори елементів-кандидатів створюються шляхом розширення великих наборів елементів іншими елементами в транзакційних даних.

Алгоритм SETM генерує набори елементів-кандидатів теж при скануванні бази даних, проте він враховує набори елементів лише в кінці сканування. Нові набори елементів-кандидатів генеруються, як і алгоритмом AIS, але ідентифікатор транзакцій, що їх генерує, зберігається з набором елементів-кандидатів в

послідовній структурі даних. Після проходу створюється лічильник підтримуваних наборів елементів-кандидатів методом агрегування послідовної структури. Професор Саед Саяд стверджує, що недоліком алгоритмів AIS і SETM є невелика кількість згенерованих і підрахованих наборів елементів-кандидатів.

Алгоритм Apriori створює набори елементів-кандидатів з використанням великих наборів минулого проходу. Великий набір елементів минулого етапу об'єднується сам з собою, щоб створити всі набори елементів, розмір яких на один більший. Кожен згенерований набір елементів з невеликою підмножиною потім видаляється. Інші набори елементів є кандидатами. Алгоритм Apriori розглядає будь-яку підмножину набору елементів, що часто зустрічається, такою як часто зустрічається. Професор Саяд стверджує, що так алгоритм скорочує кількість кандидатів, досліджуючи лише ті набори елементів, чия затребуваність більше мінімальної [10].

1.5 Питання якості новини і вибору джерела інформації

Якісні новини важливі не тільки самі по собі, але в першу чергу через їх політичне значення. Проте визначити та виміряти якість засобів масової інформації важко, бо критерії оцінки залежать від уявлення про ідеальне суспільство, які не є узгодженими. Існує думка, що з часом якість ЗМІ лише погіршується, однак скарги на низьку якість не нові. Уже 100 років тому були скарги, що газети не виправдовують високого рівня довіри, покладеного на них. Нещодавно було визначено, що у світі постійно зростає кількість політичних новин, однак їх масова частка в ЗМІ знижується. Ситуація з кількістю розважальних новин різниться між країнами: десь вона збільшується, десь залишається на сталому рівні. Хоч нема доказів зменшення розмаїття ЗМІ, проте суспільство стикається з тим, що ЗМІ є в чиїйсь приватній власності й з цього випливає дезінформація, заангажованість та упередженість у викладенні інформації.

Однак саме цей нюанс і виявляє відсутність чіткості в понятті «якість ЗМІ», бо воно визначається дуже по різному, наприклад за співвідношенням серйозних і розважальних новин, формою власності чи рівнем упередженості. Загалом, джерела за темою якості ЗМІ розмірковують про багато аспектів (актуальність інформації, збалансованість, ясність та інші) при цьому не звертаючи увагу на операційність та вимірність. Емпіричний аналіз отримує більшу точність, зосереджуючись на конкретних типах ЗМІ чи послуговуючись конкретними індикаторами, як-от кількість важливих новин, але схильний не використовувати теоретичні основи. Відсутність єдиного обов'язкового методу визначення, операціоналізації та вимірі якості ЗМІ можна пояснити подвійною герменевтикою Гіденса. Його теорія встановлює фундаментальну різницю між природничими та соціальними науками. Коли природознавчі дисципліни вивчають зовнішню матерію, тоді соціальні науки вивчають явища, в тому числі якість ЗМІ, які вже інтерпретуються суспільством по різному. Тому в суспільних науках неможливо отримати єдині й загальновизнані причини визначення по двох взаємопов'язаних причинах. Найперше, соціологи інтерпретують наперед інтерпретовані явища. Окрім того інтерпретації, що вступають в публічний дискурс, модифікують інтерпретації людей, як наслідок змінюючи початковий предмет дослідження. Це Гіденс і називає подвійною герменевтикою.

Визнаючи подвійну герменевтику, потрібно розглядати роль соціології в динамічному, випадковому та суперечному характері конструкту і прагнути до таких визначень та критеріїв, що будуть відкриті для вивчення, нової інтерпретації та перебудови [11]. Якість новинних ЗМІ – особливий тип якості. Щоб визначити межі якості необхідно визначити сам термін «якість ЗМІ». Згідно з оксфордським словником англійської мови, якість – це стандарт чогось у порівнянні з іншими об'єктами подібного роду. Інше визначення з цього ж словника: якість – ступінь переваги чогось. Ці два визначення показують, що якість ЗМІ – відносна

конструкція, і її можна позначити як об’єкт, ідеал, клас і критерій. На рисунку 1.7 представлено схему визначення якості ЗМІ.

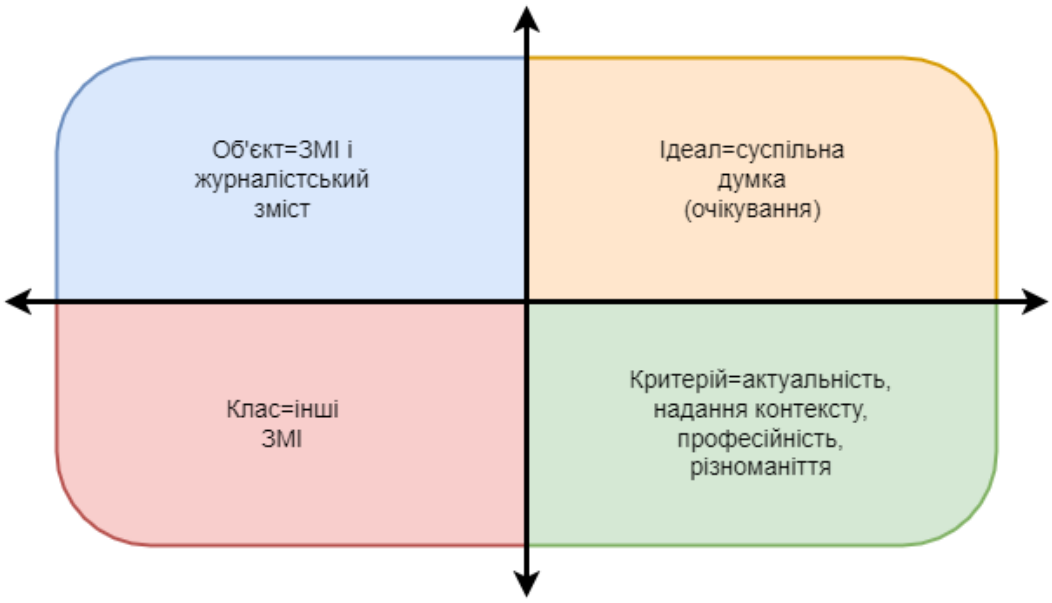

Рисунок 1.7 – Схема визначення якості ЗМІ

Об’єкт належить до класу, тому медіасистему можна оцінювати принаймні осмислено лише в порівнянні з іншими медіасистемами. Так само можна оцінювати конкретний ЗМІ лише у порівняння з іншими ЗМІ. Стосовно ідеалу виникає питання, з якої нормативної точки зору оцінюється якість ЗМІ. У найзагальнішому сенсі ЗМІ мають сприяти покращенню суспільства. Деякі вчені стверджують про суспільну цінність ЗМІ. Оскільки розуміння якості ЗМІ базується найчастіше на теорії демократії, тому і часто ідеал ЗМІ у вкладі у функціонування динамічного, демократичного суспільства. Проблема в тому ж що існують різні моделі демократії й з цього знову ж виникають труднощі з оцінюванням якості ЗМІ [12].

Дотримуючись традиції ліберальної демократії найчастіше називають такі параметри якості ЗМІ – прийнятність, різноманітність, законність, актуальність, професіоналізм. Або їх подають в розширеному вигляді: актуальність, повнота, збалансованість, ясність, різноманітність, фактологічність, нейтральність, об’єктивність, професіоналізм, правдивість.

Вказана чотирикомпонентна схема визначення є надто загальною, щоб її використовувати та вимірювати якість новинних медіа в конкретному контексті. Тому потрібно уточнити розуміння якості ЗМІ, звужуючи межі чотирьох елементів: об'єкта, ідеалу, класу і критерію. Вважається, що ідеальна модель нарадчої демократії очікує, що політична публічна сфера забезпечить формування різноманітних суспільних думок. Щоб це забезпечити, система повинна бути саморегульованою, незалежною від політично-економічного контролю. Лише це дозволить їй висвітлювати інформовані дискурси й погляди громадянського суспільства. У вершині змісту, продукованого журналістами, повинен знаходитися розум. Тоді можна розглядати новинні ЗМІ, що представляють спільний інтерес і їх зміст як об'єкт якості ЗМІ.

Неформальна ієрархія базується на критеріях цінності дискурсу. Журналістський контент повинен бути релевантним, тобто створюватися, відбиратися і поширюватися згідно з принципом надання переваги загальним темам перед приватним і суспільним змістом перед приватним. Тобто перевага в темах новин повинна віддаватися політиці, бізнесу, науці, технологіям, а не новинам про знаменитостей, спорт чи захоплення. Журналістський зміст повинен бути насичений контекстом, позаяк публічний дискурс виграє у порівнянні зі ЗМІ, які не просто повідомляють новини, але найперше вміщують інформацію у більш широкий суспільно-політичний контекст. Окрім надання фактів, ЗМІ мають пояснювати події та надавати їм контексту. Виробництво контенту має відповідати високим журналістським стандартам: прагненню до об'єктивності, прозорості та перевірки. Також зміст новин має бути різноманітним, оскільки без багатьох думок навіть найкраща ідея не буде розвиватися. Підсумовуючи, якість новинних медіа означає, що ЗМІ і його журналістський зміст (об'єкт) перевершують інші ЗМІ чи програють їм в створенні політичного публічного середовища, що забезпечує формування багатьох зважених обґрунтованих суспільних думок (ідеальних) у

порівнянні з іншими ЗМІ (клас) і оцінюються з точки зору актуальності, надання контексту, професіоналізму та різноманітності (критерії) [13].

Розглянемо детальніше кожен із критеріїв окремо. Актуальність новин – це ідеал, який, як вже було сказано, вважає загальні питання пріоритетнішими, ніж окремі, а соціальні питання більш пріоритетними, ніж приватні. Актуальність – двокомпонентний аспект, що містить в собі актуальність теми й актуальність актора. За зменшенням актуальності розташовуються такі теми: політика, економіка, культура, спорт та захоплення. При цьому, наприклад, тема спорту більш актуальна, ніж тема людських захоплень, бо може сприяти інтеграції та єднанню суспільства. Стосовно актора – ключової особи новин, його значущість теж впливає на актуальність новини. Так найактуальнішими є новини, які стосуються суспільства в цілому, або ж його великих функціональних систем. На високому рівні актуальності новини про організації та різноманітні інституції. Далі за рейтингом новини, які стосуються людей, які виконують певні функціональні ролі, а лише потім новини про особистісні аспекти людини. Причому, актуальність новини – це не середнє арифметичне обидвох показників, а величина більш наближена до середнього геометричного [14].

Засоби масової інформації повинні не просто висвітлювати події. Тому варто виділити два аспекти надання контексту новині. Найперше, новинний контент повинен вибудовувати події в довгострокові події та контексти, тобто надавати користувачеві новин довгострокову довідкову інформацію. По-друге, ЗМІ мають бути джерелом орієнтування, тобто надавати інтерпретацію подій. Тобто важливішим у питанні контексту є тематичний репортаж, у порівнянні з епізодичним, оскільки саме перший виконує тематичну класифікацію подій і вибудовує причинно-наслідкові зв’язки. Епізодичні матеріали повідомляють про одиничні події, не вбудовують їх в контекст і тому є менш цінними.

З точки зору інтерпретації подій визначення починається з жанру (формату) новини й вказує наскільки новина допомагає у формуванню думок. Новинні сюжети й репортажі, в основі яких дослідження, інтерпретація та аналіз, а також формати новин, орієнтовані на думку, такі як коментарі та редакційні статті, в яких надано й обґрунтовано точку зору є найбільш цінними. Наступне місце в рейтингу надання контексту займають інтерв'ю та новинні репортажі, складені журналістами. Новина погано інтерпретована, якщо є слабко відредагованою перепублікацією з іншого джерела чи тим більше є копією іншого тексту [15].

Критерій професіоналізму належить до соціально та демократично обґрунтованих стандартів якості, який базується на самосприйнятті професійної інформаційної журналістики. Об'єктивність змінних, прозорість джерел і незалежна звітність є тими індикаторами, які визначають професіоналізм новини чи видання. Параметр об'єктивності показує панівний стиль аргументації новини. Повідомлення в когнітивно-нормативному стилі відповідає демократичним принципам, оскільки усі аргументи мають бути об'єктивно зіставлені один з одним. Значно менш об'єктивними є ті новини, що написані в емоційному стилі (полемічні), тобто негативно впливають на раціональний дискурс. Визначання прозорості джерела інформації належить до професійної вимоги журналістів повідомляти джерела, що будуть використані для створення новини. Новина є прозорою, якщо містить пряме посилання на джерело. Джерелом може бути ім'я автора, абревіатура чи посилання на інформаційну агенцію.

Незалежна звітність дає високу оцінку якості ЗМІ, якщо звіт фокусується на зовнішніх послугах, як от копіювання агенції. Журналістика може використовувати свої демократичні функції лишень тоді, як вона є незалежною від зовнішніх послуг комунікативного постачання. Тобто найбільш цінними є новини власних кореспондентів. Репортажі інших співробітників (сумісників) теж цінуються. Тексти запрошених авторів чи експертів ідуть наступними в рейтингу. Менш

цінними є матеріали підготовані спільно з іншою редакцією. Найгіршими є новини скопійовані з інформагенцій і відредаговані чи навіть не відредаговані.

На відміну від трьох попередніх критеріїв, які визначаються на рівні конкретних новин, різноманітність характеризується загалом для усіх репортажів ЗМІ. Тобто різноманітність є якісним виміром, що проявляється не в кожній окремій новині, але в сумі всіх новин конкретного ЗМІ. Знову ж таки різноманітність – двокомпонентний параметр, що включає різноманітність контенту та географічну різноманітність.

Різноманітність контенту передбачається шляхом комбінації актуальності теми, актора та надання контексту (тематичної орієнтації). Вважається, що 50% новин мають бути присвячені різноманітним аспектам політики, при цьому допускається незначна варіативність у розподілі на внутрішньо та зовнішньополітичні теми. Чверть новин повинна бути на тему економіки, з них половина тему макроекономіки. Культура і мистецтво, а також спорт і захоплення мають охоплювати приблизно по 12 відсотків від усіх новин.

Географічна різноманітність вимірює ступінь охоплення новинами різних географічних регіонів. Для регіональних видань це новини місцевого/регіонального значення, державні національні новини та новини взаємодії конкретної країни з іншими країнами, новини іноземних держав та міжнародні (багатонаціональні) новини. Оскільки найчастіше досліджуються ЗМІ національно значення, тому для них упускається категорія місцевих/регіональних новин. Тому в ідеалі ЗМІ національного рівня повинні приділяти однакову частку всім трьом географічним регіонам у висвітленні новин. Знову ж таки значення різноманітності у числовій оцінці є середньоквадратичним різноманітності контенту та географічної різноманітності [16]. Повна схема критеріїв якості на рисунку 1.8.

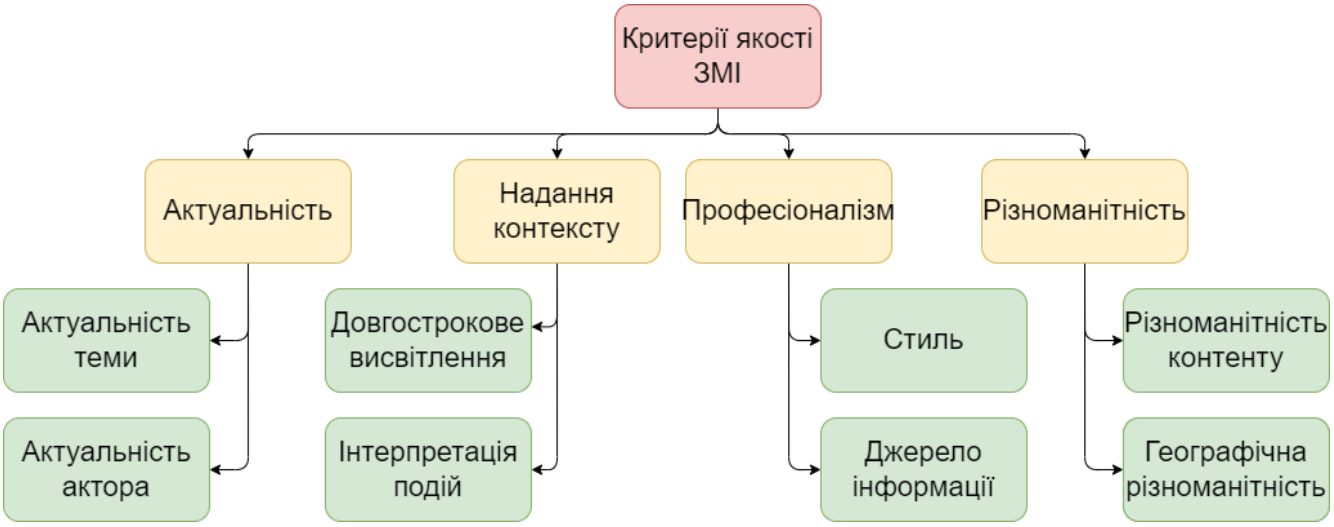

Рисунок 1.8 – Критерії якості ЗМІ

В сучасному контексті втрати доходів і заходів щодо скорочення витрат якість ЗМІ є важливою темою, що безпосередньо та опосередковано впливає на політику, економіку та культуру. Тому перевірені виміри якості важливі для оцінки стану медіасистем. На жаль, цей процес не є простим через подвійну герменевтику соціальних наук. Способи, якими соціологи оцінюють якість новинних ЗМІ, постійно взаємодіють зі значеннями конструкцій, що розділяють суспільство. Це робить якість новинних медіа динамічною, умовною та суперечливою конструкцією. А все-таки, при встановленні чотирьох основних компонентів якості ЗМІ (об’єкт, ідеал, клас і критерії) можна отримати досить зрозумілу та адекватну систему оцінки ЗМІ.

Пропонується коротко розглянути найякісніші та найвпливовіші англомовні новинні сайти, які потрапляють в більшість рейтингів, в тому числі складених за наведеною вище методологією.

New York Times вважається найвпливовішою газетою у США. Критикується за ліві прогресивні погляди, проте навіть абсолютною більшістю критиків визнається високий рівень етичних стандартів та класичних елементів журналістики. New York Times тривалий час вважається новинним об’єднанням,

що формує порядок денний. Є лідером у сфері висвітлення бізнесу, політики та культури.

Wall Street Journal – газета з найбільшим тиражем у США, спершу зарекомендувала себе як спеціаліст у діловій сфері, проте з 2007 року перекваліфікувалася та освітлює новини загальної тематики. Є більш консервативним виданням у порівнянні з New York Times, за що шанується прибічниками республіканської партії, але, однак, є впливовим виданням у різних колах.

Washington Post завжди ділила місця в трійці найбільших національних газет США з точки зору отримання журналістських премій, підготовки сенсаційних матеріалів та найму кращих репортерів. Чи не найактивніше працює в онлайн середовищі, користується підтримкою Джозефа Безоса. Є центристським виданням, з незначним нахилом вліво.

BBC – перший представник у кожному рейтинг не американських видань. BBC – британський глобальний стандарт в радіо- та тележурналістиці. Американськими рейтингами оцінюється більш як центристське видання, можливо через певний британський консерватизм. Economist – інший британський журнал, що отримує найвищі позиції в усіх рейтингах новин. Найбільшою причиною критики Economist є відсутність підписів авторів статей.

New Yorker оцінюється як креативне видання, оскільки приділяє важливу увагу науково-популярним статтям, окрім того, публікує художню літературу, точніше те, що сам так визначає. Відрізняється чи не найвищим рівнем виваженості публікацій та ретельним вивченням теми, надає прогресивний погляд на світ. Критикується за те, що та отримує компліменти – виваженість і певну неспішність.

Associated Press, Reuters, Bloomberg News – інформаційні агенції структура і принцип роботи яких відрізняється від попередніх новинних сайтів. Вони надають читачам інформацію зібрану з різних джерел іншими ЗМІ чи їх власними

кореспондентами. За високі вимоги до публікованих новин та дотримання журналістських стандартів теж завжди потрапляються в рейтинги найякісніших ЗМІ.

Foreign Affairs – чи не найсерйозніше видання в усіх рейтингах. Видається Радою з міжнародних відносин, фокусується на питаннях зовнішньої політики. Atlantic – ще одне американське новинне видання, яке надає погляд на новини США і світу з Вашингтону. Його інформують провідні журналісти світу, що є авторами детальних статей та різностороннього аналізу. Видання дотримується принципів фактологічної журналістики, але критикується за клікабельність заголовків. Politico – також один з ключових гравців у сфері політичних репортажів у США. Включається в більшість рейтингів [17].

Залежно від принципів оцінювання та обраних методологій часто, але рідше ніж вище наведені новинні сайти, в рейтинги найякісніших виробників новин потрапляють такі видання: National Public Radio, TIME, Los Angeles Times, USA Today, CNN, NBC News, CBS News, ABC News. Надпотужними й авторитетними, але особливо в галузі бізнесу та економіки є FORBES, Fortune, Financial Times. Популярними і якісними виданнями, які все ж таки є дуже політично направленими є National Review, Weekly Standard – правих поглядів, та New Republic, Nation – лівих поглядів. Узагальнення наяйкісніших видань представлено на рисунку 1.9.

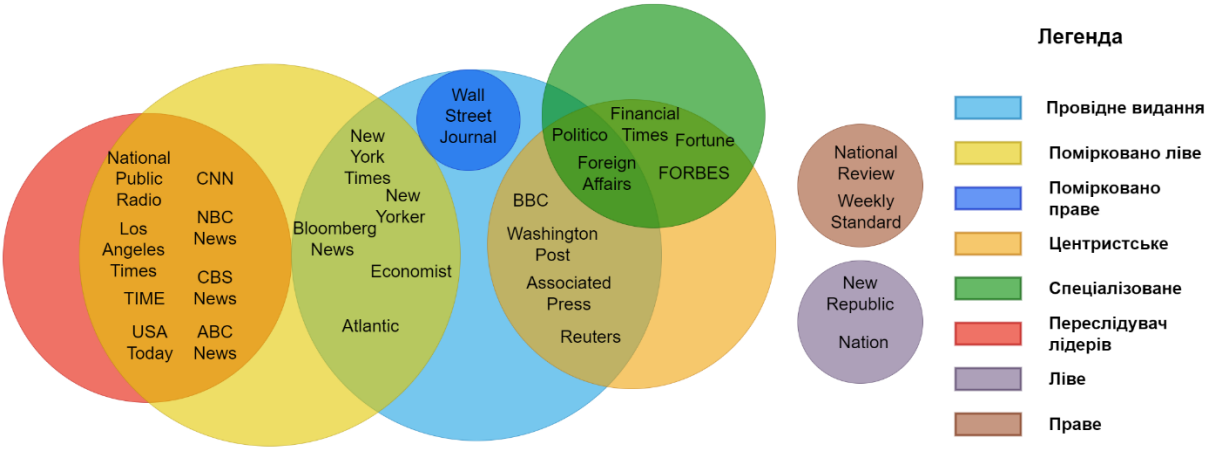

Рисунок 1.9 – Діаграма Ейлера-Вена найякісніших новинних ЗМІ

Зважаючи на наведені вище факти та рейтинги для аналізу новин та прогнозування подій на їх основі бажано використовувати новини отримані з вказаних джерел інформації.

1.6 Огляд наукових робіт пов’язаних з прогнозуванням подій

Загалом, аналіз новин для прогнозування майбутніх подій – тема малодосліджена через свою складність. Серед іншого в роботі [18] за допомогою методів обробки природної мови досліджувались особливості виникнення взаємопов’язаних сенсаційних новин на основі аналізу тексту. У статті досліджувалися закономірності появи пар подій, що настали, у просторі новин і як спрогнозувати, що інша подія настане після першої. Методи комп’ютерної лінгвістики використовувалися, щоб знайти причинно-наслідкові зв’язки між подіями з їх тестових описів.

Дослідження [19] присвячене темі виявлення причинно-наслідкових зв’язків між подіями в соцмережах, щоб спрогнозувати тональність події та час між настанням різних подій. Спершу вибираються повідомлення за проміжок часу, з них вибираються ключові слова, що використовуються для визначення тональності

повідомлення – позитивної, негативної чи нейтральної. Для визначення тональності слів використовується класифікатор, що навчається на основі методу опорних векторів. Прогнозування подій виконується з використанням часового аналізу повідомлень та розрахунку причинно-наслідкових зв’язків.

Робота [20] бере за основу лінгвістичний аналіз і статистичне моделювання твітів для автоматичного визначення тем, що обговорюються у великих містах. Щоб виокремити теми, рекомендується застосовувати тематичне моделювання, для цього своєю чергою текст твітів розбивався на токени з допомогою спеціального токенізатора та часткового мовного тегера. При цьому смайлики розглядалися як окремі токени, що несуть певне змістовне навантаження. Для моделювання теми також аналізується семантичний контент, що описує емоційний стан автора твіта.

Проблема впливу заголовків новин на поведінку інвесторів та зміни на фінансовому ринку висвітлена у роботі [21]. Модель, що заснована на зважених асоціативних правилах, визначає чи достатньо важливою є новина для інвесторів. Під час навчання на реальних даних алгоритм зважених асоціативних правил знаходить терміни, що часто й одночасно з’являються в заголовках новин. Термін з’являється в заголовках новин за день кілька раз в певний день упродовж певної кількості днів. І вага впливу терміну визначається через те, як сильно змінилася ціна акції за період з врахуванням частоти вживання даного терміну. Ці ваги дозволяють визначити чи впливають ті чи інші терміни на результати торгівлі.

Робота [22] пропонує методи, що розв’язують проблему ідентифікації подій, що є передвісниками та власне передбачень майбутніх подій. За даними колекції потокових новин з відкритих джерел був розроблений вкладений підхід для прогнозування значних публічних подій та протестів. Сильні сторони вказаного підходу доводяться емпіричною оцінкою, що полягає в фільтрації потенційних передвісників для точного прогнозування характеристик подій громадянських заворушень і в самому прогнозування настання подій з перевагою в часі виконання.

Автори дослідження [23] представляють модель прогнозування нещасних випадків, що закінчилися смертю, а також стихійних лих. Автори зібрали текстові повідомлення з системи Google про катастрофи. Отримані текстові документи оброблялися за допомогою методів комп'ютерної лінгвістики й хибні результати відсіювалися за допомогою навченого баєсівського класифікатора. Після збору даних була проведена семантична кластеризація цих даних. Матриця переходів була побудована з ключових слів, які використовувалися при зборі даних. Матриця спостереження ж була збудована зі згрупованих подій. Обидві матриці подавалися на вхід прихованої марковської моделі для прогнозування. Щоб спрогнозувати нову подію із вказаною темою, потрібно створити модель її формування на основі опису її часового ряду, а потім знайти функцію щільності розподілу її параметрів. При складанні прогнозів основна проблема аналізу та моделювання часового ряду полягає в тому, що в будь-який момент часу існує лише одна реалізація процесу (одна статистична вибірка, одна вибірка часового ряду, який вже реалізовано), які потрібно використати для створення прогнозу на майбутнє. Незалежно від того, які інструменти використовуються в методі аналізу: статистичні моделі, нейронні мережі чи моделі нечіткої логіки, нестаціонарний часовий ряд розбивається на окремі області, де він є квазістаціонарним зі своєю вибірковою функцією розподілу, і є частини ряду, в яких відбуваються перехідні процеси. Тривалість перехідного процесу визначається як фізичними змінами, так і розміром вибірки, що використовується для статистичного аналізу. Параметри функції розподілу з'ясовуються на основі аналізу даних на часовому інтервалі квазістаціонарності. Зокрема непараметричні методи допомагають відновити щільність імовірності на основі спостережуваних значень. Практично виникає дві проблеми: визначити часовий інтервал квазістаціонарності та визначити початок перехідного періоду з мінімальною затримкою.

Для прогнозування новин, що настануть, доцільно розглянути часові залежності в потоках подій і ввести кусково-постійну апроксимацію їх інтенсивності, використавши для цього баєсівський підхід та розподіл Пуассона для опису майбутніх подій.

1.7 Висновки до розділу

У розділі розглянуто загальні підходи та надана схема розв’язання задачі аналізу новин та прогнозування подій на їх основі. Це такі етапи: попередня обробка тексту, визначення ключових слів, кластеризація новин за тематикою, оцінка кластеризації, аналіз новин і побудова прогнозу. Проведено огляд підходів щодо кластеризації текстових даних, визначено переваги та сфери застовування різних алгоритмів. Вирішено використовувати агломеративний алгоритм та метод к-середніх.

Визначено різноманіття підходів до аналізу та прогнозування текстових даних із виокремленням їх сильних та слабких сторін. Зазначено, що може використовуватися різноманітний математичний апарат, проте асоціативні правила видалися чи не найбільш відповідними.

Зауважено, що важливим аспектом є питання вибору джерела новин, тому проаналізовано питання якості новин. Визначено, які ЗМІ подають інформацію найкраще. Вирішено використовувати як вхідні дані лише новини із якогось з названих ЗМІ.

Проведено огляд наукових робіт за темою прогнозування подій.

Отже, варті уваги методи кластеризації даних, а саме метод к-середніх та агломеративна кластеризація у зв’язку з їх доступністю та широтою поширення. Для прогнозування доцільно скористатися асоціативними правилами, запропонувавши їм нове застосування, на відміну від стандартих кошиків користувача та прогнозування покупок.

2 ОПИС МЕТОДУ РОЗВ'ЯЗАННЯ ЗАДАЧІ ТА РОЗРОБКА АЛГОРИТМУ

2.1 Змістовна постановка задачі

На основі набору новин англійською мовою (текстових документів) розробити алгоритм, що будуватиме ланцюги, які відображатимуть причинно-наслідкові зв'язки між подіями.

2.2 Формалізована постановка задачі

Нехай $X = \{X_1, X_2, \dots, X_n\}$ – набір подій. Розглянемо відношення вигляду $R = \{\text{подія } X_1 \text{ спричиняє подію } X_2\}$ або ж у вигляді $R = \{X_i \to X_j\}$. Дане відношення є антирефлексивним, асиметричним і транзитивним, інакше кажучи відношенням строгого порядку. Необхідно побудувати ланцюг подій вигляду $X_i \to X_j \to \cdots \to X_l$, де $X_i, X_j, X_l \in X$.

2.3 Опис методів попередньої обробки даних

Оскільки текст напряму не може бути вхідними даними для кластеризація чи побудови асоціативних правил, то необхідно виконати його попередню обробку. У дисертації застосовано зокрема різні підходи підготовки вхідних даних. Найперше виконано очищення тексту усіх небуквенних символів, зокрема цифр і спеціальних символів. Згодом виконано розбиття тексту статті на токени (слова). Стоп-слова, які не містять у собі змісту чи містять його мінімальну кількість, необхідно вилучити. Використано стоп-слова англійської мови, що задано в переліку стоп-слів Natural Language Toolkit. Окрім того, перелік стоп-слів розширено за допомогою визначених самостійно стоп-слів, що наведені у додатку В. Опісля необхідно провести лемматизацію токенів, для цього доречно використати WordNetLemmatizer. WordNetLemmatizer виконує лемматизацію послуговуючись для цього WordNet. WordNet – це семантична мережа (словник) для англійської

мови, розроблена вченими Прінстонського університету. Характерною особливістю цього словника є те, що базовою одиницею є не окреме слово, як у звичних словниках, а синонімічний ряд, який об’єднує слова із близьким значенням, що є вузлами мережі. Кожен синонімічний ряд доповнений визначенням і прикладами використання його слів у контексті. Слово чи сполука слів може з’являтися декілька разів в одному синонімічному ряді і мати декілька частин мови. Додатково кожен ряд містить список синонімів і вказівників, що описують відношення між ним та іншими рядами [24]. По завершенню лемматизації слова потрібно назад об’єдати в уже очищений текст статті. Блок-схема описаного вище алгоритму наведена на рисунку 2.1.

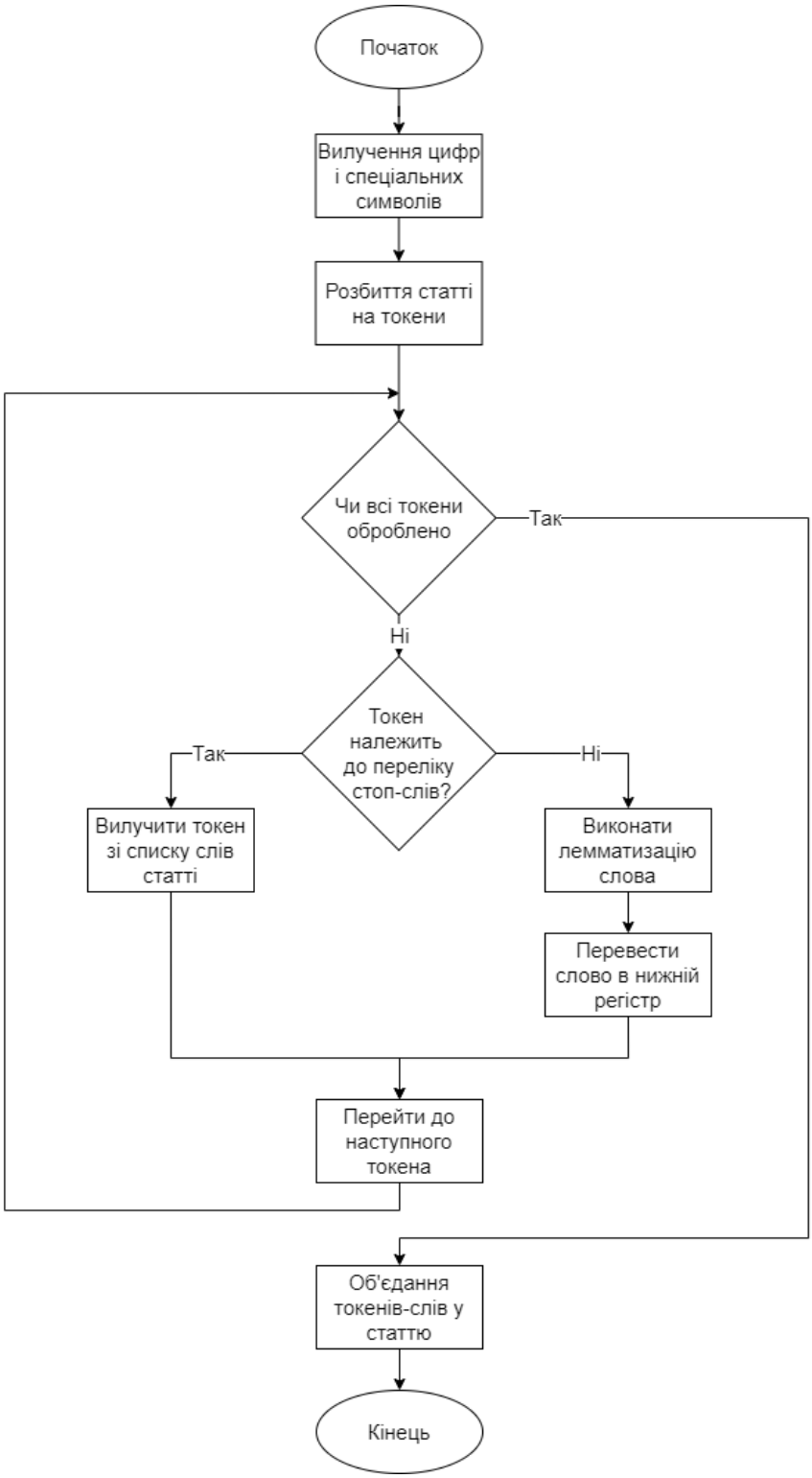

Рисунок 2.1 – Блок-схема алгоритму очищення і стандартизації текстових даних

Наступним кроком після очищення тексту від усього зайвого є його трансформація у вигляд, який може опрацьовувати алгоритм кластеризації. У цьому допоможе TF-IDF векторизація (метод визначення частоти й зворотної частоти документу). Оскільки більшість алгоритмів послуговуються математичними структурами, як вхідними даними, наприклад, числами, матрицями і таким іншим, тому тексти природною мовою потрібно перетворити у вектор, саме це перетворення і називається векторизацією. Це ключовий процес обробки тексту. TF-IDF – текстовий векторизатор, що перетворює текст у вектор. Він поєднує 2 принципи – частоту терміну TF і частоту документа DF. Частота терміну – кількість зустрічей певного слова в документі. Вона вказує наскільки важливим є це слово в документі. Частота термінів представляє кожен текст у вигляді матриці (таблиці), де рядок – кількість документів, а стовпці – кількість окремих термінів у всіх документах [25].

Частота документів – кількість документів, що містять певний термін. Частота документів вказує, як сильно поширений той чи інший термін. Зворотня частота документа (IDF) – вага терміну. Необхідно зменшити вагу терміну, якщо його входження розкидане по всіх документах. IDF – розраховується так:

$$idf_i = \log\left(\frac{n}{df_i}\right), \tag{2.1}$$

де idf_i – значення IDF для терміна i, n – загальна кількість документів, df_i – кількість документів, що містять термін i. Чим вище DF терміна, тим нижче його значення IDF. Коли кількість DF дорівнює n, то це поняття міститься у всіх документах, тоді IDF буде дорівнювати нулю. Отже, цей термін не несе багато інформації. Оцінка TF-IDF отримується множенням частотної матриці терміна на IDF і можна подати у такому вигляді:

$$w_{i,j} = tf_{i,j} * idf_i, \tag{2.2}$$

де $w_{i,j}$ – оцінка TF-IDF для поняття i в документі j, $tf_{i,j}$ – частота поняття для поняття i в документі j, а idf_i – оцінка IDF для терміна i.

Пропонується розглянути простий приклад для кращого розуміння. Нехай існує три простих тексти. Текст 1 – відбудеться другий тур президентських виборів в Україні (його нормалізована версія – відбуватися другий тур президентський вибори в Україна). Текст 2 – в Києві відбулися дебати Ющенка і Януковича (в Київ відбуватися дебати Ющенко і Янукович). Текст 3 – на президентських виборах в Україні переміг Ющенко (на президенський вибори в Україна перемагати Ющенко). Рисунки 2.2 – 2.4 показують кроки обрахунку матриці TF-IDF.

Крок 1. Частотна матриця

	відбуватися	другий	тур	президентський	вибори	в	Україна	Київ	дебати	Ющенко	і	Янукович	на	перемагати
Текст 1	1	1	1	1	1	1	1							
Текст 2	1					1		1	1	1	1	1		
Текст 3				1	1	1	1			1			1	1

Рисунок 2.2 – Частотна матриця

Крок 2. Вектор IDF

	відбуватися	другий	тур	президентський	вибори	в	Україна	Київ	дебати	Ющенко	і	Янукович	на	перемагати
IDF	0.17609	0.4771	0.4771	0.17609	0.17609	0	0.17609	0.4771	0.4771	0.17609	0.4771	0.4771	0.4771	0.4771

Рисунок 2.3 – Вектор IDF

Крок 3. Перемноження частотної матриці і вектора IDF

	відбуватися	другий	тур	президентський	вибори	в	Україна	Київ	дебати	Ющенко	і	Янукович	на	перемагати
Текст 1	0.17609	0.4771	0.4771	0.17609	0.17609	0	0.17609	0	0		0	0	0	0
Текст 2	0.17609	0	0	0		0		0.4771	0.4771	0.17609	0.4771	0.4771	0	0
Текст 3	0	0	0	0.17609	0.17609	0	0.17609	0	0	0.17609	0	0	0.4771	0.4771

Рисунок 2.4 – Матриця TF-IDF

Іншим і трохи складнішим способом представлення вхідних даних для алгоритмів кластеризації є використання моделі FastText. Це безкоштовно бібліотека із відкритим вихідним кодом, що використовується для вивчення природних мов. Авторами моделі є Facebook AI Research. FastText дозволяє створити алгоритм навчання з вчителем чи без вчителя для отримання векторного представлення тексту. FastText швидко навчає векторні моделі слів, на відміну від

моделей з використанням глибоких нейронних мереж, де використовуються лінійні класифікатори. Модель FastText вирішує проблему швидкодії використовуючи ієрархічний класифікатор. У цьому випадку він представляє мітки в двійковому дереві. Кожен вузол в двійковому дереві представляє ймовірність. Мітка представлена ймовірністю на шляху до цієї мітки. Це означає, що листок двійкового дерева представляє мітку. FastText послуговується алгоритмом Хафмана для побудови двійкових дерев, щоб врахувати можливість того, що вони не збалансовані. Глибина міток, що зустрічаються часто менша, ніж тих, що зустрічаються рідко. Використання двійкового дерева пришвидшує час пошуку, оскільки не потрібно перглядати всі вузли, а досить лише шукати вузли. Завдяки цьому змешується швидкість роботи алгоритму.

Ключовий принцип FastText полягає в розгляді кожного слова як скупності підслів. Щоб було простіше і не було залежності від мови, підслова – це символьні n-грами слова. Вектор для слова є сумою всіх векторів його символьних n-грам.

У роботі використовується обидва підходи, оскільки агломеративні алгоритми кластеризації витрачають надто багато часу для обробки матриці TF-IDF.

2.4 Опис методів кластеризації у задачі кластеризації текстових новин

Саме кластеризація дозволяє розділити набір текстів на категорії. Пропонується розглянути детально методи кластеризації, що будуть використані в роботі для кластеризації текстів новин.

Основну увагу приділимо ієрархічним алгоритмам кластеризації. В ієрархічних алгоритмах не задається чітко кількість кластерів чи умова зупинки кластеризації.

Нехай існує набір даних $D = \{x_1, \dots, x_n\}$, де $x_i \in R^d$ і кластеризація $\mathbf{C} = \{C_1, \dots, C_k\}$, що розділяє множину D. Кожен кластер є множиною векторів $C_i \in D$, причому кластери попарно не перетинаються $C_i \bigcap C_j \ (\forall\, i \neq j)$ і $\bigcup_{i=1}^{k} C_i = D$. Кластеризація $A = \{A_1, \dots, A_r\}$ називається вкладеною в іншу кластеризацію $B = \{B_1, \dots, B_s\}$, якщо $r > s$ і $\forall\, A_i \in A\ \exists\, B_i \in B$ такий що, $A_i \subseteq B_j$. Ієрархічна кластеризація дає послідовність вкладених множин $C_1, \dots, C_n$, розпочинаючи з тривіальної кластеризації $C_1 = \{\{x_1\}, \dots \{x_n\}\}$, де кожен елемент знаходиться в окремому кластері до іншої тривіальної кластеризації $C_n = \{\{x_1, \dots, x_n\}\}$, де всі елементи знаходяться в одному кластері. В загальному випадку кластеризація C_{t-1} вкладена в кластеризацію C_t [26].

Розглянемо детальніше алгоритм агломеративної ієрархічної кластеризації:

Алгоритм агломеративної ієрархічної кластеризації

Вхід: D **Вихід:** $\mathbf{C}$
$\mathbf{C} \leftarrow \{C_i = \{x_i\} \mid x_i \in D\}$ //кожна точка лежить в окремому кластері
$\Delta \leftarrow \{d(x_i, x_j)\colon x_i, x_j \in D\}$ //визначити матрицю відстаней
Поки $|\mathbf{C}| \neq k$
    Знайти найближчу пару кластерів: $C_i, C_j \in \mathbf{C}$
    $C_{ij} \leftarrow C_i \cup C_j$ //об'єднання кластерів
    $\mathbf{C} \leftarrow \mathbf{C} \setminus \{\{C_i\} \cup \{C_j\} \cup \{C_{ij}\}$ //оновлення кластеризації
    Оновити матрицю відстаней Δ, щоб провести нову кластеризацію

Ключовим елементом алгоритму є пошук найближчої пари кластерів, тобто вибір критерію подібності. Як правило, найчастіше використовується метрика відстаней.

Дивізивна ієрархічна кластеризація менш поширена ніж агломеративна, хоч працює за аналогічним принципом як і агломеративна, але в протилежному напрямку. У роботі не розглядається застовання дивізивних алгоритмів

кластеризації, натомість використовується додатково неієрархічний алгоритм, а саме метод к-середніх [27].

Отже, нехай існує припущення щодо кількості кластерів k – тобто задана їх кількість. Серед множини об'єктів випадковим чином обирається k об'єктів як початкові центри кластерів. Визначається відстань від кожного об'єкту до центру кластеру і на основі цього задається приналежність об'єкту до кластеру з мінімальною відстанню. Потім визначається новий центр кластеру як середнє за кожною ознакою об'єктів, що на попередній ітерації віднесено до кластера. Загалом, в результаті застосування методу к-середніх множина об'єктів розділяється на k кластерів, розташованих на максимальній відстані один від одного.

Нехай $U = \{u_{ij}\}, i = 1 \dots n, j = 1{,}2$ – матриця приналежності об'єкта до певного кластеру. Стовпець 1 вказує індекс кластеру, до якого належить об'єкт i, а стовпець 2 – відстань від об'єкта до центру кластера. $C^{(0)} = \left\{c_l^{(0)}\right\}, l = 1 \dots k$ – матриця координат центрів кластерів. $Q^{(0)}$ – велике число (функціонал якості), ε – точність обчислення функціоналу, m – номер ітерації.

Розглянемо детально схему побудови кластерів методом к-середніх:

Алгоритм кластеризації методом к-середніх

Вхід: D **Вихід:** U
$C^{(0)} \leftarrow \{x_1, x_2, \dots, x_k\}, U^{(0)}, Q^{(0)}, \varepsilon, m = 1$ //ініціалізація параметрів
$\Delta \leftarrow \{d(x_i, x_j): x_i, x_j \in D\}$ //розрахунок відстаней до центрів кластерів
$U^{(m)}, Q^{(m)}$ //визначення матриці приналежності та функціоналу
Поки $|Q^{(m)} - Q^{(m-1)}| > \varepsilon$
    $C^{(m)}$ //розрахунок нових центрів кластерів
    $Q^{(m-1)} \leftarrow Q^{(m)}$
    $m \leftarrow m + 1$
    $\Delta \leftarrow \{d(x_i, x_j): x_i, x_j \in D\}$ // відстані до центрів кластерів $C^{(m-1)}$
    $U^{(m)}, Q^{(m)}$ //визначення матриці приналежності та функціоналу

19 Побудова кластерів за $U^{(m)}$

Щоби визначити кількість кластерів, на яку необхідно розділити набір новинних статей доцільно скористатися двома методами: метод ліктя (elbow method) та методом силуету (silhouette method).

Метод ліктя є чи не найвідомішим підходом визначення оптимальної кількості кластерів. Проте він має певний недолік – наївність у підході. Метод пропонує обчислити внутрішньокластерну суму квадратів помилок для різної кількості кластерів та обрати ту, при якій на графіку помилок буде видно лікоть [28].

Загалом внутрішньокластерну суму квадратів помилок можна представити так:

$$WSS = \sum_{l=1}^{k} \sum_{i \in S_l} d^2(a_i, c_l), \tag{2.3}$$

де $l -$ номер кластера, $l = 1 \dots k$, k – кількість кластерів, S_l – множина об’єктів в кластері l, $d^2(a_i, c_l)$ – квадрат відстані від об'єкта a_i до центру кластера c_l. Для визначення відстані можна скористатися будь-якою метрикою, як-от, наприклад евклідовою чи Міньковського.

Більш елегантним є метод силуету, який визначає наскільки точка схожа на свій власний кластер (згуртованість) у порівнянні з іншими кластерами (розділення). Значення коефіцієнта силуету може коливатися в діапазоні $[-1; 1]$. Високе значення вказує на те, що об’єкт знаходиться в правильному кластері. Якщо багато точок мають від’ємне значення коефіцієнта силуету, це означає, що задана кількість кластерів надто мала чи велика.

Значення коефіцієнта силуета для кожного об’єкта з набору обраховується так:

$$s(i) = \begin{cases} \dfrac{b(i) - a(i)}{\max\{a(i), b(i)\}}, & \text{якщо } |S_i| > 1 \\ 0, & \text{якщо } |S_i| = 1 \end{cases}, \tag{2.4}$$

де $a(i)$ – міра подібності об'єкта до її власного кластера, $b(i)$ – міра відмінності об'єкта від об'єктів в інших кластерах, а $|S_i|$ – потужність кластера, в який входить об'єкт i [29].

Міри подібності та відмінності розраховуються так:

$$a(i) = \frac{1}{|S_i| - 1} \sum_{j \in S_i, i \neq j} d(i,j), \tag{2.5}$$

$$b(i) = min_{i \neq j} \frac{1}{|S_j|} \sum_{j \in S_j} d(i,j), \tag{2.6}$$

де $d(i,j)$ – відстань між об'єктами i та j. Як правило використовується евклідова відстань.

Розглянемо деякі метрики далі:

$$d_1(x_i, x_j) = \sqrt{\sum_{l=1}^{n} (x_{i,l} - x_{j,l})^2}, \tag{2.7}$$

$$d_2(x_i, x_j) = \sum_{l=1}^{n} |x_{i,l} - x_{j,l}| \tag{2.8}$$

$$d_3(x_i, x_j) = \max_{1<l<k} |x_{i,l} - x_{j,l}| \tag{2.9}$$

$$d_4(x_i, x_j) = \frac{1}{2} \sum_{l=1}^{n} \frac{(x_{i,l} - x_{j,l})^2}{(x_{i,l} + x_{j,l})}, \tag{2.10}$$

де d_1 – евклідова відстань, d_2 – метрика міста, d_3 – відстань Чебишева, d_4 – відстань хі-квадрат, а $x_i = (x_{i,1}, \dots x_{i,n}), x_j = (x_{j,1}, \dots x_{j,n})$ – об'єкти в просторі з кількість ознак n [30].

З метою візуалізації виконаної кластеризації необхідно методом головних компонент виділити 2 компоненти для кожної статті, щоб можна було подати об'єкт як точку у двовимірному просторі.

Аналіз головних компонентів – метод зменшення розмірності, який часто використовується для зменшення розмірності великих наборів даних шляхом перетворення великого набору змінних в менший, що містить більшу частину інформації великого набору. Звичайно, неможливо не втратити частину інформації зменшуючи розмір вектора вхідних даних, але метод пропонує пожертвувати трохи точністю заради простоти.

Аналіз головних компонент можна розбити на 5 кроків.

Крок 1. Стандартизація змінних – виконується з метою того, щоб кожна змінна робила однаковий вклад в аналіз, оскільки метод є чутливим до діапазонів вхідних змінних і змінні з більшим діапазоном домінуватимуть над змінними з меншим. Математично стандартизація записується так:

$$x_{is} = \frac{(x_i - \bar{X})}{\sigma_X}, \tag{2.11}$$

де $\bar{X}$ – середнє значення змінної, а σ_X – стандартне відхилення ознаки.

Крок 2. Знаходження коваріаційної матриці потрібне для того, щоб зрозуміти, як змінні набору вхідних даних відрізняються від середнього по відношенню один до одного. Тобто необхідно перевірити, чи є якийсь зв'язок між змінними.

Крок 3. Знаходження власних векторів та власних чисел коваріаційної матриці. Головні компоненти – це нові змінні, що побудовані як лінійні комбінації

вихідних змінних. Ці комбінації будують так, щоби головні компоненти не були корельовані між собою і щоби вони вміщали якнайбільше даних. Знайдені власні вектори необхідно впорядкувати за їх власними числами в порядку спадання.

Крок 4. Побудова вектора ознак. Упорядковані власні вектори дозволяють знайти головні компоненти в порядку значущості. На цьому кроці ми обираємо чи зберігати всі компоненти чи відкидати менш значущі. І формувати матрицю векторів (вектор ознак).

Крок 5. Перебудова даних по осях головних компонент. Необхідно використати вектор ознак, що був створений з власних векторів коваріаційної матриці, щоби переорієнтувати дані з вихідних осей на ті, що представлені головними компонентами. Для цього потрібно помножити транспонований вхідний набір даних на транспонований вектор ознак [31].

Щоб оцінити близькість і подібність побудованих кластерів використовується коефіцієнт Жаккара. Для цього в кожному з побудованих кластерів визначається 300 термінів з найвищим значенням TF-IDF. Міра подібності коефіцієнтом Жаккара для кластерів A та B визначається так:

$$K_{1,-1} = \frac{n(A \cap B)}{n(A \cup B)}, \tag{2.12}$$

де $n(A \cap B)$ – кількість слів в перетині слів кластерів A та B, а $n(A \cup B)$ – кількість слів в об'єданні слів кластерів A та B.

2.5 Опис методів побудови асоціативних правил

Для розв'язання задачі прогнозування подій використовуємо алгоритми побудови асоціативних правил.

Розглядаємо асоціативні правила вигляду $X_i \to X_j$, де X_i, X_j – події. Оскільки X_i, X_j – реальні події, то X_i – тотожно істинне, X_j – тотожно істинне, то імплікація

вигляду $1 \to 1 = 1$ (за властивістю імплікації). Це говорить про те, що асоціативні правила є тотожно істинними.

Асоціативні правила знаходять всі набори елементів, що мають затребуваність, що перевищує мінімальну. Потім використовують великі набори елементів для створення бажаних правил рівень впевненості яких перевищує мінімальний рівень впевненості. Ліфт правила – це відношення впевненості, що спостерігається, до очікуваної впевненості, якби X_i та X_j були б незалежними.

$$supp(X_i) = \frac{|\{t \in T; X \sqsubseteq t\}|}{|T|} \tag{2.13}$$

$$conf(X \to Y) = \frac{supp(X_i \cup X_j)}{supp(X_i)} \tag{2.14}$$

$$lift(X \to Y) = \frac{supp(X_i \cup X_j)}{supp(X_i) * supp(X_j)} \tag{2.15}$$

Розглянемо на прикладі алгоритми побудови асоціативних правил. Нехай існує набір таких подій отриманих із новин $T = \{\text{землетрус}, \text{руйнування}, \text{рятувальники}, \text{смерть}, \text{криза}\}$. Для зручності подальшого зображення пронумеруємо події так: землетрус – 1, руйнування – 2, рятувальники – 3, смерть – 4, криза – 5.

В базі даних існують такі поєднання: $x_1 = \{1,3,4\}, x_2 = \{2,3,5\}, x_3 = \{1,2,3,5\}, x_4 = \{2,5\}$.

Алгоритм Apriori

Крок 1. Набори елементів-кандидатів створюються з використанням тільки великих наборів елементів попереднього етапу без врахування поєднань в базі даних.

Крок 2. Великий набір елементів попереднього проходу об’єднується сам з собою, для створення наборів елементів, розмір яких більший на 1.

Крок 3. Кожен згенерований набір елементів, що має невелику підмножину, видаляється. Набори елементів, що залишилися є кандидатами [32].

Розглянемо приклад роботи алгоритму Apriori на рисунку 2.5.

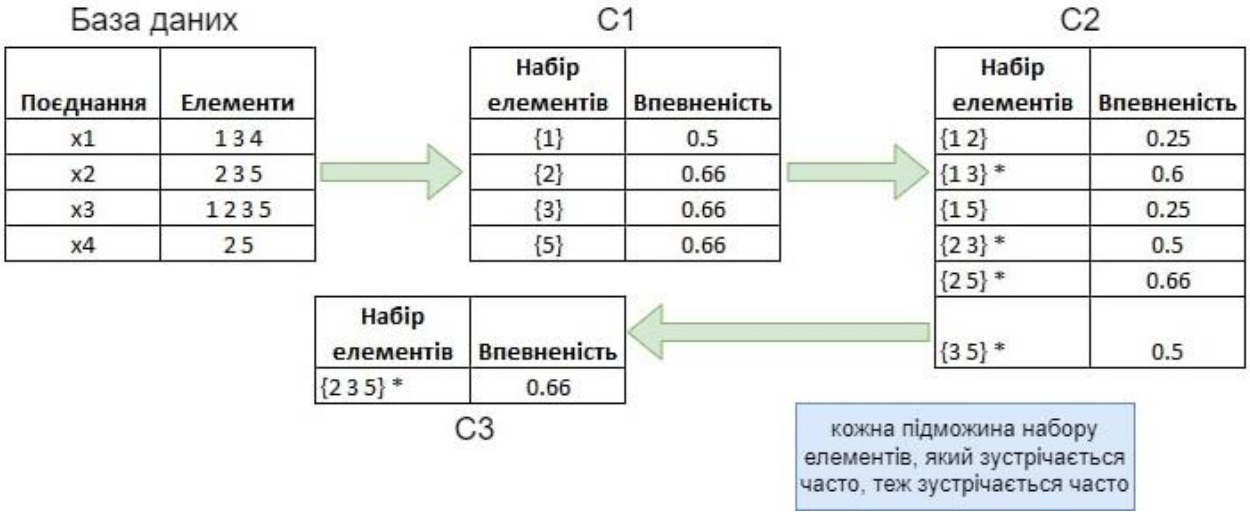

Рисунок 2.5 – Робота алгоритму Apriori

Алгоритм Apriori користується тим, що будь-яка підмножина набору елементів, що часто зустрічаються, також є таким набором, що часто зустрічається. Тому алгоритм може зменшити кількість кандидатів, яких він розглядає, досліджуючи лише ті набори, для яких впевненість перевищує мінімальну. Всі рідкісні набори елементів можуть бути скорочені, якщо в них є рідкісні підмножини.

Модифікований алгоритм Apriori

Крок 1. База даних не використовується для підрахунку впевненості елементів-кандидатів після першого проходу.

Крок 2. Набори елементів-кандидатів генеруються так само як в алгоритмі Apriori.

Крок 3. Створюється додатковий набір C', кожен член якого має ідентифікатор кожного поєднання і більші набори елементів, що присутні в цьому

поєднанні. Цей набір використовується для підрахунку підтримки кожного набору елементів-кандидатів.

Як працює модифікація алгоритму Apriori з використанням ідентифікатора поєднання можна побачити на рисунку 2.6.

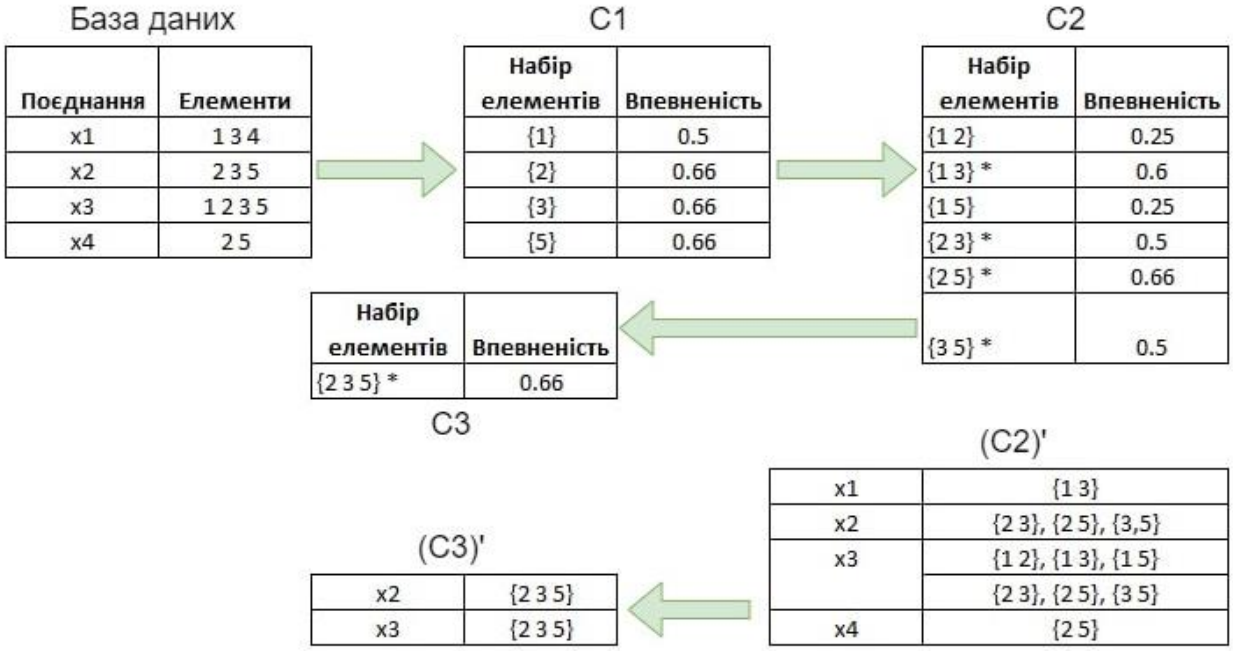

Рисунок 2.6 – Робота модифікованого алгоритму Apriori

Перевага модифікації полягає в тому, що кількість записів в C' може бути меншою, ніж поєднань в базі даних, особливо на пізніх етапах.

Доцільним може бути використання гібридного алгоритму Apriori, який працює як Apriori на початкових етапах, пізніше використовується модифікації, коли очікується, що пам’ять зможе вмістити C'.

Алгоритм FP Growth

Цей алгоритм є вдосконалення алгоритму Apriori. Частий шаблон генерується без потреби генерації кандидатів. Алгоритм FP Growth представляє базу даних у вигляді дерева, що називається деревом частих шаблонів (FP). Ця деревоподібна структура підтримуватиме зв'язок між наборами елементів. База

даних фрагментована з використанням одного частого елемента. Ця частина називається «фрагмент шаблону». Аналізуються набори елементів цих фрагментованих шаблонів. Так з допомогою цього методу пошуку наборів елементів, що часто зустрічаються, скорочується.

Як вже було сказано, дерево FP – деревоподібна структура, що створена з початкових наборів елементів бази даних. Ціль дерева FP – знайти найбільш часто вживаний шаблон. Кожен вузол дерева представляє елемент з набору елементів.

Кореневий вузол є нулем, а нижні вузли представляють набори елементів. Зв’язок вузлів з нижніми вузлами, тобто наборів елементів з нижніми наборами елементів, зберігається на етапі формування дерева.

Покроково алгоритм FP Growth можна представити так:

Крок 1. Сканування бази даних з метою знаходження входження наборів елементів в базі даних.

Крок 2. Початок побудови дерева FP. Створення кореня дерева з нульовим значенням.

Крок 3. Повторення сканування бази даних і перевірка поєднань. Знайти в першому поєднанні набір елементів. Набір елементів з максимальною кількістю береться наверх, далі з меншою кількістю і так до кінця. Це означає, що гілка дерева побудована з наборами елементів поєднань за зменшенням кількості.

Крок 4. Виконується перевірка наступного поєднання в базі даних. Набори елементів впорядковані за зменшенням кількості. Якщо якийсь набір елементів вже присутній в іншій гілці, тоді ця гілка поєднання буде мати спільний префікс для кореня. Це означає, що спільний набір елементів зв’язаний з новим вузлом іншого набору елементів в цьому поєднанні.

Крок 5. Окрім того, кількість набору елементів збільшується в міру того, як це відбувається в поєднаннях. Кількість як спільного вузла, так і нового

збільшується на 1 в порядку їх створення і зв'язування з відповідними поєднаннями.

Крок 6. Видобування створеного дерева. Досліджується нижній вузол разом зі зв'язками нижніх вузлів. Нижній вузол представляє довжину частотного шаблону 1. Звідси перейти по шляху в дереві FP. Ці шляхи називаються базою умовного шаблону – підбазою даних, що складається з префіксів в дереві FP, що зустрічаються з найнижчим вузлом (суфіксом).

Крок 7. Побудувати умовне дерево FP, яке формується шляхом підрахунку наборів елементів в шляху. Набори елементів, що відповідають пороговій впевненості, розглядаються в умовному дереві FP.

Крок 8. Часті шаблони генеруються з умовного дерева FP [33].

Приклад роботи алгоритму FP Growth показано на рисунку 2.7.

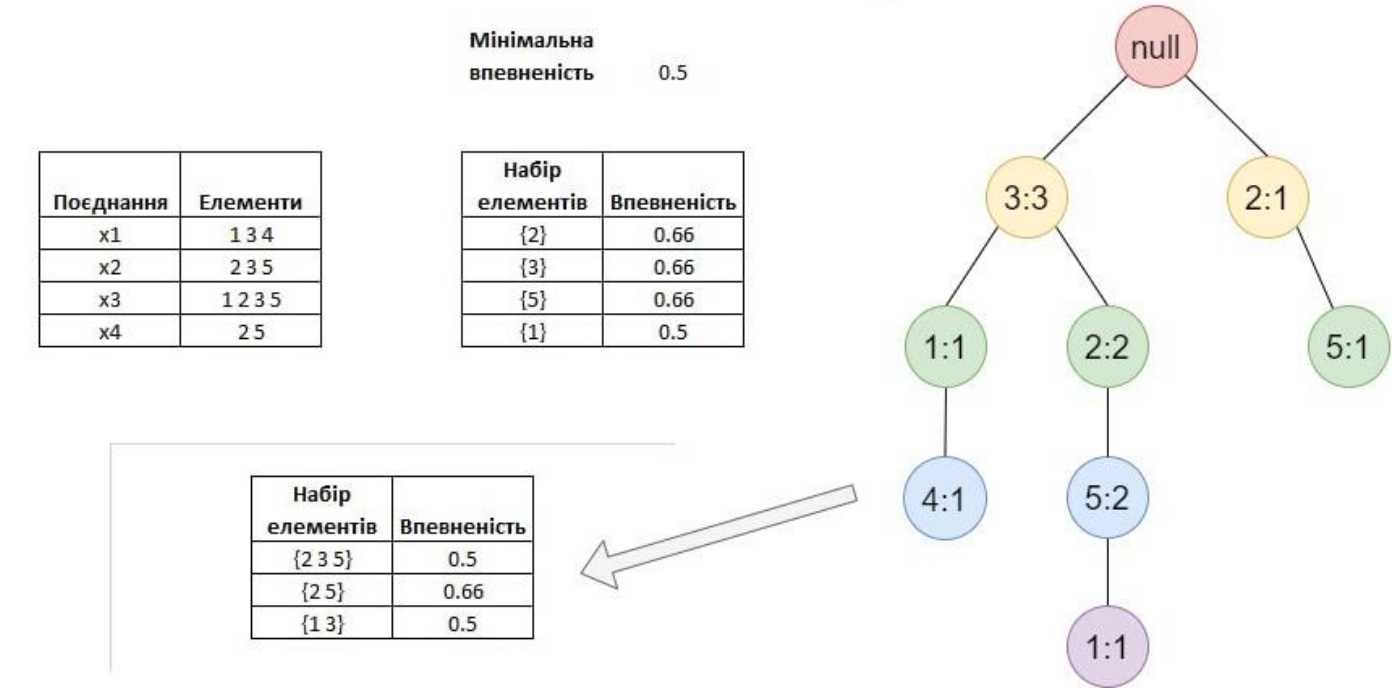

Рисунок 2.7 – Робота алгоритму FP Growth

Основними перевагами алгоритму FP Growth є те, що він сканує базу даних лише двічі, на відміну від алгоритмів Apriori, які сканують поєднання на кожній

ітерації, а також він масштабований та ефективний для пошуку часто вживаних наборів різної довжини. Недоліком алгоритму є його складність в порівнянні з алгоритмом Apriori.

Щоб отримати якісні та інтерпретовані результати побудови асоціативних правил, щоб мати змогу виявляти причинно наслідкові зв'язки виконуємо такі кроки:

Крок 1. Обираємо кластеризацію проведену методом к-середніх з використанням TF-IDF.

Крок 2. Об'єднуємо в один кластер попарно кластери з коефіцієнтом Жаккара більше 0.28.

Крок 3. Представляємо дані у вигляді необхідному для роботи алгоритму пошуку асоціативних правил FP-Growth (матриця – у якій рядок відповідає статті, а стовпець – слову, якщо слово міститься у статті, тоді на перетині рядка і стовпця стоїть 1), що у вигляді формули можна подати так:

$$E_{i,j} = \begin{cases} 1, & \text{якщо слово } j \text{ наявне у рядку } i \\ 0, & \text{інакше} \end{cases}, \qquad (2.16)$$

Крок 4. Виконуємо побудову асоціативних правил із мінімальною затребуваністю подій:

$$supp(X_i) \geq \frac{\sum_i^m \sum_j^n E_{i,j}}{n * m}, \qquad (2.17)$$

де n – кількість статей, а m – кількість слів.

При цьому мінімальний ліфт $lift(X_i \to X_j) = 1$.

Крок 5. Відфільтровуємо асоціативні правила за рівнем впевненості та затребуваності.

Крок 6. Об’єднуємо правила в ланцюги. Якщо існує побудоване асоціативне правило вигляду $X_i \rightarrow X_j$ та існує правило вигляду $X_j \rightarrow X_l$, тоді існує ланцюг $X_i \rightarrow X_j \rightarrow X_l$.

2.6 Висновки до розділу

У розділі подано змістовну та формальні постановки задачі прогнозування подій на основі новинної інформації.

Запропоновано підходи, що використовуватимуться для кластеризації новин, а саме метод кластеризації к-середніх та ієрархічний агломеративний алгоритм. Надано описи алгоритмів, що будуть використовуватися для побудови кластерів. Визначено метрики відстаней, що можнуть використовуватися при визначенні відстаней між об’єктами. Подано обгрнутування методів визначення кількості кластерів.

Для задачі прогнозування подій визначено використовувати аналіз асоціативних правил з використанням алгоритму FP-Growth, оскільки він є більш сучасною та якісною версією алгоритму Apriori.

3 ОПИС ПРОГРАМНОГО ТА ТЕХНІЧОГО ЗАБЕЗПЕЧЕННЯ

3.1 Засоби розробки

Перед початком розробки програмного продукту було виконано огляд інструментальних засобів та обрано ті, які найбільше підходять під реалізацію даної системи, є найбільш зручними та надійними. Зокрема використано такі засоби та технології:

- фреймворк Spring Boot для реалізації серверної частини;
- платформа для розробки вебзастосунків Vaadin;
- мови програмування: Java, Python;
- інтегроване середовище розробки програмного забезпечення IntelliJ IDEA Ultimate Student Edition;
- бібліотеки для мов програмування – Hibernate, nltk, mixtend, genism, Scikit-learn;
- Google Colab – хмарне середовище розробки Jupiter.

Вебзастосунок розроблено на мові програмування Java, яка паралельна, заснована на класах, об’єкто-орієнтована. Вона була зумисне розроблена, щоб мати якнайменшу кількість залежності від реалізації. Застосунки на Java можуть виконуватися на будь-якій системі чи пристрої, що використовує Java Virtual Machine. Перевагами Java є:

- мова компільована та інтерпретована: поєднано обидва підходи, компілятор перетворює Java код в інструкції байт-коду, а інтерпретатор генерує машинний код;
- можливість до переносу: зміна і оновлення операційних систем, процесорів і системних ресурсів не потребує зміни застосунку на Java. Переносимість

забезпечується двома способами: генерацією байт-коду для виконання на будь-якому пристрої і незалежністю примітивних типів даних від машини;

- об’єктна орієнтованість: майже все в Java є об’єктом, широка кількість об'єднаних в пакети класів, які можна використовувати, також об'єктну модель легко розширювати;
- безпечність і надійність: Java розроблений зі збірником сміття, щоб управляти пам’яттю, концепція обробки винятків дозволяє зменшити імовірність серйозних помилок;
- розподіленість – саме Java – перша мова програмування розрахована на роботу в мережі, застосунки на Java можуть легко отримувати доступ до інтернет-ресурсів, що означає розподіленість ресурсів між користувачами;
- багатопоточність та інтерактивність – Java підтримує здатність процесора виконувати декілька процесів чи потоків водночас, тому не потрібно чекати поки виконається те чи інше завдання, щоб виконувати інше;
- висока продуктивність – послуговуючись компіляцією Just-In-Time, збіркою сміття, багатопоточністю Java-застосунки показують високу продуктивність, цьому сприяє й архітектура мови [34].

Для шаблонної і правильної організації роботи застосунку послуговувалися Spring Boot – фреймворк, що суттєво спрощує і пришвидшує розробку веб- та мікросервісних застосунків за допомогою Java Spring Framework, при цьому конфігурація та налаштування – мінімальні. Spring Boot дозволяє уникнути написання вручну стандартного коду та складних конфігурацій. Загалом Spring – популярна платформа с відкритим кодом для розробки автономних застосунків. Саме її потужність і є причиною основого недоліку Spring – складності налаштування, настройки і розгортання застосунку. Spring Boot допомагає

розробникам швидко все налаштувати завдяки автоконфіграції, самостійному підходу та автономності застосунку [35].

Spring Boot ініціалізує застосунки попередньо встановленими залежностями, налаштовує базову платформу Spring, будь-які сторонні пакети на сонові вказаних налаштувань. Це допомагає уникнути помилок в майбутньому. Функція автоматичного налаштування суттєво економить час розробника і дозволяє швидко перейти до розробки. Spring Boot дозволяє заповнити просту вебформу, вказавши потрібні стартові пакети для застосунку. На основі цього він вибирає які залежності встановити і які значення за замовчування використати. Автономність забезпечується шляхом вбудови вебсерверу в сам застосунок, без потреби взаємодії і використання зовнішнього сервера.

Vaadin є платформою з відкритим кодом для створення сучасних вебзастосунків. Саме цю платформу взято за основу для представлення графічного інтерфейсу користувача. Vaadin об'єднує фреймворки, інструменти та компоненти користувацького інтерфейсу в один продуманий стек розробки. Vaadin призначений для створення не дуже складних, однак професійних вебзастосунків. За замовчування, всі застосунки Vaadin є прогресивними вебзастосунками, тобто користувачі можуть встановлювати їх на свої пристрої. API Flow Java дозволяє використовувати об'єктноорієнтовану модель для компонування своїх складників в їх представлення, використовуючи макети. Додатково є змога надавати типізовані сервіси для надання зовнішнього інтерфейсу за допомогою кількох анотацій в класі сервісу Java. Якщо коротко, то Vaadin призначений для підвищення продуктивності розробки нескладних вебзастосунків з можливістю зосередитися на створенні функціональності.

З метою зручної роботи з сутностями в програмі використовувався Hibernate, за потреби в майбутньому він дозволить легко і просто взаємодіяти з базою даних. Hibernate використовується для об’єктно-реляційного зіставлення об’єктів і

сутностей бази даних. Це фреймворк, що забезпечує абстракцію таких технологій як JDBC, сервлет та інші. Hibernate розроблює логіку, яка зберігає та обробляє дані для більш тривалого використання. Це простий інструмент об'єктно-реляційної проєкції з відкритим кодом. Своєю чергою, об'єктно-реляційна проєкція – метод зображення об'єкта, що зберігається в базі даних зі спрощеним доступом до нього. Для цього використовується внутрішнє API мови Java [36].

У Hibernate відсутні недоліки інших технологій. Hibernate на відміну від JDBC легко змінює дані в базі, також він підтримує перенесення коду, він укріплює відношення на рівні об'єктів, надає змогу уникнути частини обробки виключних ситуації, яка є обов'язковою для JDBC. Це зменшує довжину коду з підвищенням його читаності. Оскільки Hibernate має відкритий код, тому він доступний для кожного без жодних витрат. Hibernate – легкий фреймворк, його ефективність підвищується, якщо не використовувати контейнери. Хоча Hibernate може працювати з різними технологіями водночас, він може працювати та самостійно. Він має специфічний характер – не потрібно реалізовувати інтерфейси API, оскільки класи розробки застосунків Hibernate слабко зв'язані.

Алгоритми й методи обробки тексту виконані мовою програмування Python, адже саме вона ідеально підходить для цих цілей. Найпершою причиною цьому є простота – Python легко читається і зрозумілий для новачків. У нього акуратний і зрозумілий синтаксис, тому він допомагає якнайшвидше перейти до написання коду, не витрачаючи час на вивчення документації. Python широко використовується для аналізу даних, статистичного аналізу, веброзробки, обробки текстів, що і є основною темою цієї магістерської роботи.

Важливу роль при виборі Python складає те, що у ньому є величезна кількість бібліотек для інтелектуального аналізу різних типів даних. Дуже популярними є такі бібліоткеи, як Pandas, statsmodels, NumPy, SciPy і Scikit-Learn. Такі екосистеми як SciPy спрощують задачі з обробки та аналізу даних. SciPy задовільняє широкий

спектр загальних потреб, як-от обробка структур даних, аналіз складних мереж, алгоритми та інструменти машинного навчання. Бібліотеки Python постійно розвиваються, оскільки весь час все більша кількість розробників залучені до спільноти і роблять свій внесок у розвиток бібліотек. Прикладом є Keras – мінімалістична бібліотека для глибокого навчання. Вона стала важливим компонентом екосистеми Python [37].

Мультипарадигмальність – яскрава ознака Python. Тобто не потрібно створювати клас для виконання найпростішої операції, як от вивід шматка тексту. Використовуючи мультипарадигмальний підхід, підтримуються функціональний, процедурний, об'єктноорієнтований та аспектно-орієнтований стилі програмування. Python легко вбудовується в застосунки написані іншими мовами програмування, тому дозволяє легко інтегруватися з іншими мовами, спрощуючи написання вебзастосунків. Python надається з унікальною структурою модульного тестування, що може також виконувати різні тести для розробки складних застосунків [38].

Мабуть, кожен розробник Python знайомий із Jupyter Notebook. Це вебзастосунок із відкритим вихідним кодом, що дозволяє зручно працювати над інтелектуальним аналізом даних і машинним навчанням. Jupyter Notebook дозволяє показувати результати роботи в тому ж документі, де знаходиться програмний код. Серед багатьох сервісів пов’язаних із Jupyter Notebook виділяється Google Colaboratory, яка дозволяє безплатно виконувати хмарні обчислення і надає доступ до графічного процесора. Оскільки Google Colaboratory виконує синхронізацію з гугл диском, саме там можна зберігати дані і результати досліджень. У цій магістерській роботі також використано переваги роботи з Google Colaboratory.

Доцільно більш детально зупинитися на окремих бібліотеках Python використаних у цій магістерській. Natural Language Toolkit (NLTK) – набір інструментів чи платформа, яка працює з даними природними мовами для

застосування в статистичній обробці. Ця платформа містить бібліотеки для обробки тексту для токенізації, синтаксичного аналізу, класифікації, виокремлення коренів та іншого. Також вона містить графічні приклади та демонстрації наборів даних з поясненням базових задач обробки природних мов. Спочатку написана Стівеном Бердом, Едвардом Лопером і Юеном Кляйном для використання в розробці та навчанні. Детальні інструкції та приклади роблять NLTK зручною для лінгвістів із малими знаннями в програмування, а також для всіх дослідників, яким потрібно заглибитися в комп'ютерну лінгвістику [39].

Scikit-learn – чи не найкорисніша бібліотека для машинного навчання у мові Python. Вона містить інструментарій для проведення машинного навчання і статистичного моделювання, в тому числі класифікацію, кластеризацію, регресію та зменшення розмірності. Ця бібліотека використовується для побудови моделей машинного навчання, вона не призначання для читання та інших маніпуляцій з даними. Scikit-learn підтримує величезну кількість алгоритмів, зокрема такі алгоритми навчання з учителем: узагальнені лінійні моделі, метод опорних векторів, дерева рішень, баєсівські класифікатори. До того, Scikit-learn надає змогу виконати перехресну перевірку на невидимих даних і оцінити точність розроблених моделей. Забезпечується і робота з великою кількістю алгоритми навчання без учителя – факторний аналіз, аналіз основних компонент, кластеризація (яка і була використана в даній роботі) і навіть нейронні мережі без вчителя. Окрім того, Scikit-learn підходить для отримання ознак із текстів (модель «торба слів») чи зображень.

Також у даній роботі було використано бібліотеку Gensim, зокрема для побудови моделі Fast Text. Власне Gensim – популярна бібліотека обробки природних мов із відкритим текстом, що використовується для тематичного моделювання без вчителя. Використовує академічні моделі та статистичне машинне навчання для виконання таких задач:

- побудова векторів документів чи слів;

- створення корпусу;
- індентифікація теми;
- порівняння документів;
- аналіз текстових документів на семантичну структуру.

Окрім вказаних задач Gensim також призначений для обробки великих текстових наборів з використанням потокової передачі даних. Це вигідно відрізняє його від інших пакетів машинного навчання, які здатні виконувати обробку лише в пам’яті. Вважається, що Gensim є масштабованим, бо не має потреби зберігати весь вхідний корпус постійно в оперативній пам’яті, тобто всі алгоритми Gensim не залежать від пам’яті відносно розміру корпусу [40]. Gensim надійний, бо використовується вже тривалий час, незалежний від платформи, позаяк працює на всіх платформах, що підтримують Python і Numpy. Щобільше, для деяких алгоритмів наявні ефективні багатоядерні реалізації, що пришвидшує його роботу.

Для реалізації асоціативних правил використано бібліотеку mlxtend. Це також бібліотека для машинного навчання, яка містить корисні інструменти для щоденних задач з обробки та аналізу даних. MLxtend пропонує додаткові функції машинного навчання і може використовуватися як додаткова бібліотека до вже наявного списку бібліотек [41]. Список можливостей є не дуже великим, але не вичерпується такими можливостями:

- побудова кореляційного кругу методу головних компонент;
- компроміс зсуву та дисперсії;
- побудова областей прийняття рішень моделей класифікації;
- створення матриці точкових діаграм;
- статистичний бутстреп;
- побудова асоціативних правил.

3.2 Вимоги до технічного забезпечення

Щоби забезпечити коректну роботу розробленого застосунку, необхідно пересвідчитися у наявності встановленого такого програмного забезпечення:

- Операційна система Windows 10;
- мова Python, версія 3.7;
- JDK, версія 18;
- фреймворк Vaadin, версія 23;
- бібліотека gensim, версія 4.2.0;
- бібліотека mlxtend 0.17.

Окрім цього, звісно, повинен бути встановлений вебпереглядач Chrome, Firefox чи Safari останніх версій. Мінімальна конфігурація комп'ютера повинна бути такою:

- обсяг оперативної пам'яті – не менше 4 Гб;
- тактова частота процеса – не менше 1.6 ГГц;
- наявність монітора, клавіатури, маніпулятора;
- технічні показники інших компонентів не є такими важливими.

3.3 Архітектура програмного забезпечення

Vaadin не цілком відповідає шаблону проєктування MVC, також він не заснований на MVP, однак в багатьох проєктах корпоративних застосунів використовується MVP у поєданні з Vaadin, хоч насправді нема такої критичої необхідності.

Взагалі Vaadin надає дів моделі розробки вебзастосунків: на стороні сервера чи на стороні клієнта. Беззаперечно, модель розробки, що керується сервером, є більш потужною та дозволяє розробляти застосунки виключно на стороні сервера з використанням клієнтського механізму Vaadin, який показує інтерфейс

користувача в браузері. Клієнтська модель дозволяє розробляти віджети та застосунки на Java, які компілюються в JavaScript і виконуються в переглядачі. Обидві моделі можна комбінувати, використовуючи програмний код і сервіси. Архітектура Vaadin надає ілюстрацію зв'язку на стороні клієнта та на стороні сервера, коли сторінка з кодом клієнта завантажена в браузер. Фреймворк Vaadin містить API на стороні клієнта та сервера, багато компонентів і віджетів користувацького інтерфейсу з обох сторін, а також тем для керування зовнішнім виглядом. Міститься у Vaadin і модель даних, що дозволяє напряму прив'язувати компоненти на стороні сервера до даних [42].

Хочеться трохи детальніше зупинитися на візуальній складовій архітектури застосунку. Застосунки Vaadin надають інтерфейс для користувача, щоб він мав змогу взаємодіяти з бізнес-логікою та даними застосунку. На технічному рівні інтерфейс реалізовується як клас, що розширює com.vaadin.ui.UI. Його ключовою задачею є створення початкового інтерфейсу з компонентів і налаштування подій для обробки користувацького вводу. Кожен компонент інтерфейсу на стороні сервера має аналог на стороні клієнта, віджет, який показується у вебпереглядачі і з яким взаємодіє користувач. Компоненти на стороні сервера передають події логіці застосунку. Компоненти поля, що містять значення, котрі користувач може переглядати, найчастіше прив'язані до джерела даних.

Розроблений застосунок відповідає архітектурі фреймворку Vaadin і розбитий на пакети. Перелік пакетів із їх значенням подано у таблиці 3.1.

Таблиця 3.1 – Перелік пакетів розробленого застосунку

Назва пакету	Призначення пакету
entity	Зберігання сутностей, що використовуються в застосунку
generator	Зберігання класу для завантаження статей і результатів роботи
NYT	Зберігання сутностей, пов'язаних із статтями та їх завантаженням з файлів
repository	Зберігання репозиторіїв для доступу до сутностей

service	Зберігання сервісу
views	Зберігання елементів графічного інтерфейсу (відображення)

3.3.1 Діаграма класів

Графічне зображення діаграми класів подано у додатку Б. Програма складається з великої кількості класів, нижче наведено їх короткий опис:

- ClusteringResult – клас результатів кластеризації;
- StringListConverter – клас-конвертатор для перетворення набору рядкових даних в один рядок для збереження в БД і навпаки;
- DataLoader – клас для завантаження початкових даних і результатів кластеризації в застосунок;
- NYTCorpusDocument – клас самої статті (документа);
- NYTCorpusDocumentParser – клас для парсингу статті в об’єкт з документа у форматі xml;
- ClusteringResultRepository – репозиторій об’єктів результатів кластеризації;
- NYTCorpusDocumentRepository – репозиторій, що містить всі документи (статті);
- NewsService – клас-сервіс, що надає доступ до репозиторію статей і результатів;
- ListView – представлення документів статей у табличному вигляді;
- NewsForm – представлення статті у детальному вигляді із відображенням її змісту;
- AssociationRulesView – клас-представлення побудованих асоціативних правил;
- ClusteringView – клас-представлення результатів кластеризації;
- DashboardView – клас-представлення статистичних показників;
- MainLayout – клас-представлення головного макету програми;
- Application – клас-точка входу в програму.

3.3.2 Діаграма компонентів

У додатку Б надано діаграму компонентів. Розроблений у магістерській роботі застосунок складається з таких компонентів:

- застосунок Vaadin – відповідає за роботу програми; отримує вказівки з вебсторінок і напряму звертається до джерела даних, щоби отримати потрібні відомості;
- джерело даних – містить інформацію, потрібну для роботи застосунку (вхідні документи, відомості про результати кластеризації);
- вебсторінки – графічне відображення програми, з ними взаємодіє користувач.

3.3.3 Специфікація функцій

Створена система містить величезну кількість методів та функцій, пояснення усіх не є практично корисним, тому варто переглянути специфікацію основних функцій. У таблиці 3.2 наведено специфікацію функцій вебзастосунку.

Таблиця 3.2 – Специфікація методів вебзастосунку

Клас	Прототип функції	Сементика функції	Параметри функції	Семантика параметрів
NYTCorpusDocument	public String toString()	Подання об’єкта у вигляді рядка		
NYTCorpusDocument	private void appendProperty(StringBuffer sb, String propertyName, Object propertyValue)	Додання властивості до рядкового представлення об’єкта	StringBuffer sb, String propertyName, Object propertyValue	Рядковий буфер; ім’я властивості; значення властивості

Продовження таблиці 3.2

Клас	Прототип функції	Сементика функції	Параметри функції	Семантика параметрів
NYTCorpusDocumentParser	public NYTCorpusDocument parseNYTCorpusDocumentFromFile(File file, boolean validating)	Парсинг документа з файлу	File file, boolean validating	Файл; прапорець чи виконувати валідацію
NYTCorpusDocumentParser	public NYTCorpusDocument parseNYTCorpusDocumentFromDOMDocument(File file, Document document)	Парсинг документа з DOM документа	File file, Document document	Файл; документ
NYTCorpusDocumentParser	private Document loadNonValidating(File file)	Завантаження документа без валідації	File file	Файл

Продовження таблиці 3.2

Клас	Прототип функції	Сементика функції	Параметри функції	Семантика параметрів
NYTCorpusDocumentParser	private Document loadValidating (File file)	Завантаження документа з валідацією	File file	Файл
NYTCorpusDocumentParser	private Document parseStringTo DOM(String s, String encode, File file)	Парсинг рядка в DOM документ	String s, String encoding, File file	Рядок; декодер; файл
NYTCorpusDocumentParser	private Document getDOMObject(String filename, boolean validating)	Парсинг файла в DOM документ	String filename, boolean validating	Ім’я файла; прапорець чи проводити валідацію
NYTCorpusDocumentParser	private String getAttributeValue(Node node, String attributeName)	Отримання значення атрибута	Node node, String attributeName	Вузол; ім’я атрибута

Продовження таблиці 3.2

Клас	Прототип функції	Сементика функції	Параметри функції	Семантика параметрів
NYTCorpusDocumentParser	private String getAllText(Node node)	Отримання всього тексту	Node node	Вузол
NYTCorpusDocumentParser	private List<Node> getNodesByTagName(Node node, String tagName)	Отримання вузлів за назвою тегу	Node node, String tagName	Вузол; назва тегу
NYTCorpusDocumentParser	private void recursiveGetNodesByTagName(Node node, String tagName, List<Node> matches)	Рекурсивне отримання вузла за назвою тегу	Node node, String tagName, List<Node> matches	Вузол; назва тегу; список вузлів

Продовження таблиці 3.2

Клас	Прототип функції	Сементика функції	Параметри функції	Семантика параметрів
DataLoader	public CommandLineRunner loadData(NYTCorpusDocumentRepository documentRepository, ClusteringResultRepository resultRepository)	Завантаження початкових даних і результатів кластеризації	NYTCorpusDocumentRepository nytCorpusDocumentRepository, ClusteringResultRepository clusteringResultRepository	Репозиторій статей (документів); репозиторій результатів кластеризації
ClusteringResult	public String toString()	Подання об’єкта у вигляді рядка		
ClusteringResult	private void appendProperty(StringBuffer sb, String propertyName, Object propertyValue)	Додання властивості до рядкового представлення об’єкта	StringBuffer sb, String propertyName, Object propertyValue	Рядковий буфер; ім’я властивості; значення властивості

Продовження таблиці 3.2

Клас	Прототип функції	Сементика функції	Параметри функції	Семантика параметрів
NewsService	public List<NYTCorpusDocument> findAllNYTCorpusDocuments(String stringFilter)	Пошук всіх документів за текстовим фільтром	String stringFilter	Рядок
NewsService	public long countNYTCorpusDocuments()	Підрахунок кількості документів		
NewsService	public Map<Integer, Long> findYearsStatistics()	Підрахунок статистики по роках		
NewsService	public Map<Double, Long> findLengthStatistics()	Підрахунок статистики по довжині тексту		

Продовження таблиці 3.2

Клас	Прототип функції	Сементика функції	Параметри функції	Семантика параметрів
ListView	private void updateList()	Оновлення списку документів на відображення		
ListView	private void configureForm()	Сконфігувати форму		
ListView	private void configureGrid()	Сконфігувати сітку відображення статей		
NewsForm	public void setNYTCorpusDocument(NYTCorpusDocument nytCorpusDocument)	Встановити документ	NYTCorpusDocument nytCorpusDocument	Документ

У таблиці 3.3 наведено специфікацію функції, пов’язаних з математичною частиною роботи.

Таблиця 3.3 – Специфікація обчислювальних функцій

Прототип функції	Сементика функції	Параметри функції	Семантика параметрів
def preprocess_text(text: str, remove_stopwords: bool) -> str	Виконання попередньої обробки тексту	text: str, remove_stopwords: bool	Початковий текст; прапорець чи проводити вилучення стоп-слів і лематизацію
def get_top_keywords(n_terms, group_clusters)	Отримання ключових слів кожного кластера	n_terms, group_clusters	Кількість слів; групуюча кластеризація
def jaccard_matrix(group_clusters)	Отримання матриці коефіцієнтів Жаккара	group_clusters	Групуюча кластеризація
def show_jaccard(matrix, cluster_map, title)	Зображення матриці коефіцієнтів Жаккара	matrix, cluster_map, title	Матриця коефіцієнтів Жаккара; зіставлення кластерів з їх назвою; заголовок

Продовження таблиці 3.3

Прототип функції	Сементика функції	Параметри функції	Семантика параметрів
def PlotClusterWordCloudArray(articles, articleCentroidIds, Ks, cluster_name, title)	Побудова хмари слів	articles, articleCentroidIds, Ks, cluster_name, title	Текст очищених статей; кластер статті; для яких кластерів будувати хмару; зіставлення кластерів з їх назвою; заголовок
def CreateWordCloud(text)	Створення об'єкта WordCloud з тексту	text	Текст
def ConcatinateClusterTexts(articles, articleCentroidIds, K)	Об’єднання всіх статей з одного кластера	articles, articleCentroidIds, K	Текст очищених статей; кластери статей; кластер для якого виконувати об'єднання
def CountClusterArticles(articles, articleCentroidIds, K)	Підрахунок кількості статей у кластері	articles, articleCentroidIds, K	Текст очищених статей; кластери статей; кластер для якого виконувати об'єднання

Продовження таблиці 3.3

Прототип функції	Сементика функції	Параметри функції	Семантика параметрів
def PlotWordCloud(text)	Графічне зображення створеної хмари слів	text	Текст
def K_determination(points)	Визначення найкращої кількості кластерів методами ліктя та силуету	points	TF-IDF матриця
def pca_calculation(arr_to_transform, comp_number)	Визначення головних компонент	arr_to_transform, comp_number	Матриця для обробки; кількість компонент
def averaged_word2vec_vectorizer(corpus, model, num_features)	Побудова матриці із заданою кількістю ознак	corpus, model, num_features	Токенізовані статті; модель FastText; кількість ознак

Продовження таблиці 3.3

Прототип функції	Сементика функції	Параметри функції	Семантика параметрів
def merge_near_clusters(DF, matrix)	Об’єдання статей з найближчих кластерів	DF, matrix	Датафрейми зі статттями розбитими по кластерах; матриця коефіцієнтів Жаккара
def prepare_for_rules(DFF)	Перетворення статей кластерів у дані для побудови асоціативних правил	DFF	Датафрейми зі статтями
def build_association_rules(dataset, min_supp, metric_name, metric_value)	Побудова асоціативних правил	dataset, min_supp, metric_name, metric_value	Вхідні дані; мінімальна підтримка для частих шаблонів; метрика для побудови правил; мінімальне значення метрики

Продовження таблиці 3.3

Прототип функції	Сементика функції	Параметри функції	Семантика параметрів
def rules_normalization(rules)	Нормалізація асоціативних правил для побудови ланцюга правил	rules	Правила
def graph_building(rules_graph)	Побудова графа правил	rules_graph	Нормалізовані правила
def find_paths(g, rules_graph) -> list	Пошук ланцюга правил	g, rules_graph	Граф; нормалізовані правила
def find_text_paths(paths, key_val) -> list	Побудова ланцюга в текстовому вигляді	paths, key_val	Ланцюги; ключі для слів
def print_rules_graphical(g)	Побудова ланцюга у вигляді графа	g	Граф

3.4 Керівництво користувача

Щоби працювати з розробленим вебзастосунком користувачеві потрібно відкрити вебпереглядач та перейти за адресою http://localhost:8080/. Найперше користувач побачить головну сторінку «Новини», вона показана на рисунку 3.1.

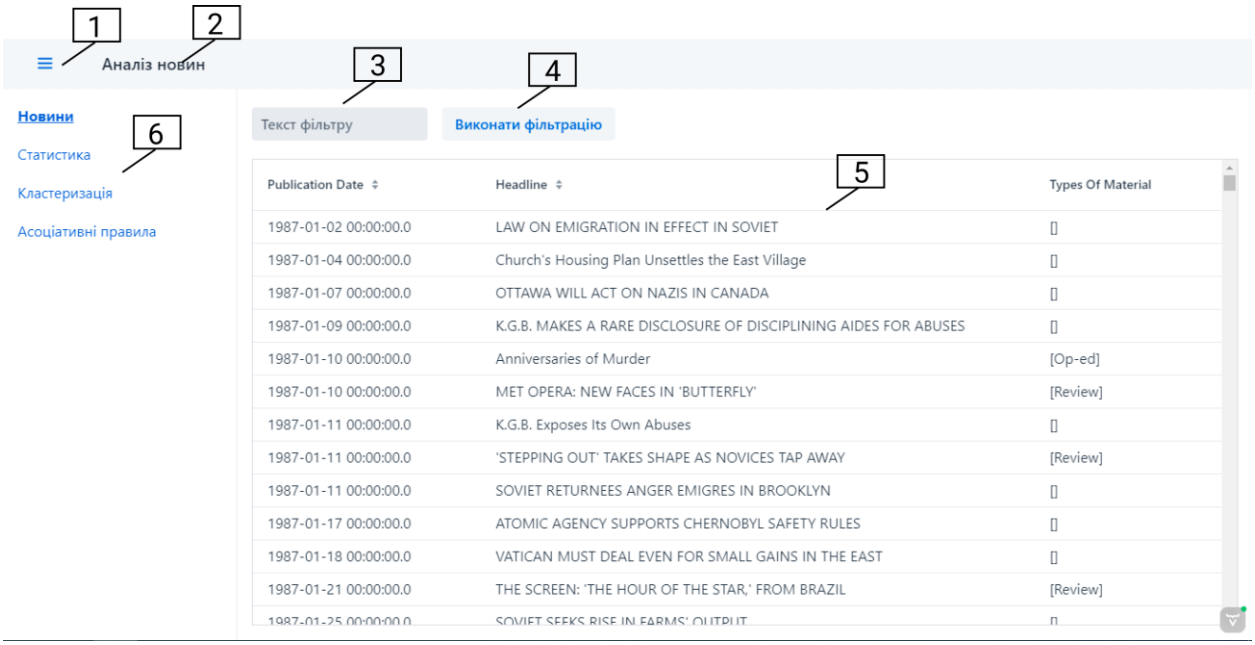

Рисунок 3.1 – Головна сторінка застосунку

Користувач має змогу бачити такі поля та елементи:

– елемент 1 – кнопка згортання-розгортання меню;
– елемент 2 – коротка назва системи;
– елемент 3 – поле для фільтрації статей за датою публікації чи фрагментом назви;
– елемент 4 – кнопка для фільтрації;
– елемент 5 – сітка статей з такими колонками «Дата публікації», «Загаловок», «Тип матеріалу»;
– елемент 6 – меню застосунку, містить навігацію між сторінками «Новини», «Статистика», «Кластеризація», «Асоціативні правила».

Уведемо в елемент 3 число 2006 і система відобразить нам статті за 2006 рік, про що можна пересвідчитися на рисунку 3.2.

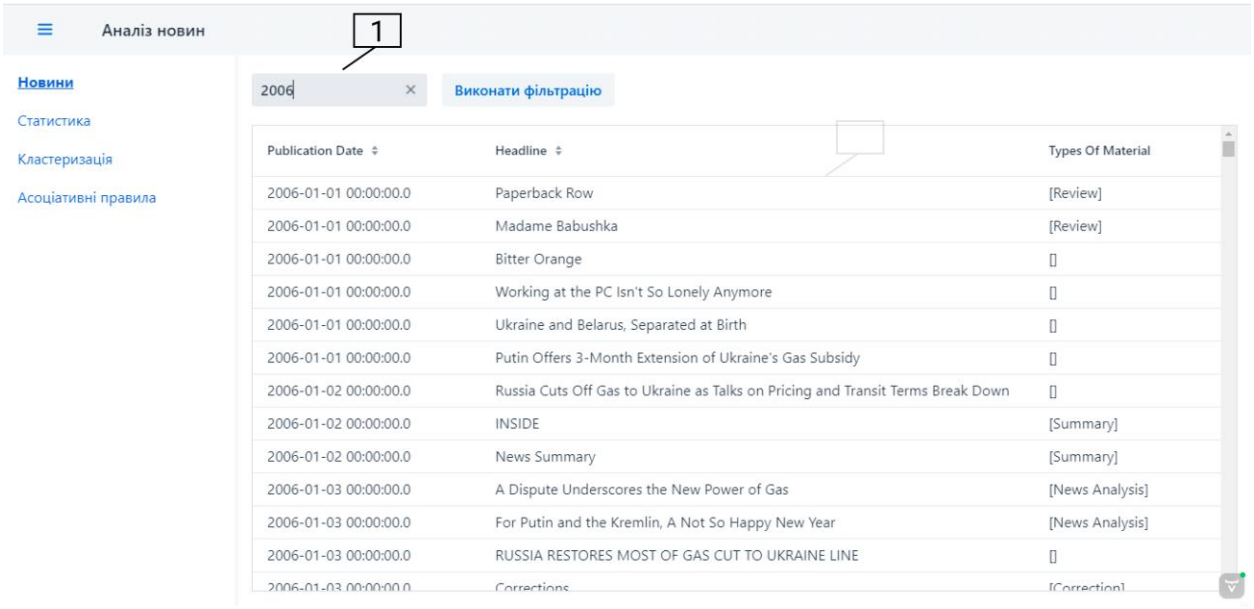

Рисунок 3.2 – Головна сторінка з новинами відфільтрованими за датою

Уведемо в елемент 3 слово «Clinton» і система покаже нам статті, які містять в заголовку слово «Clinton». Про це можна пересвідчитися на рисунку 3.3.

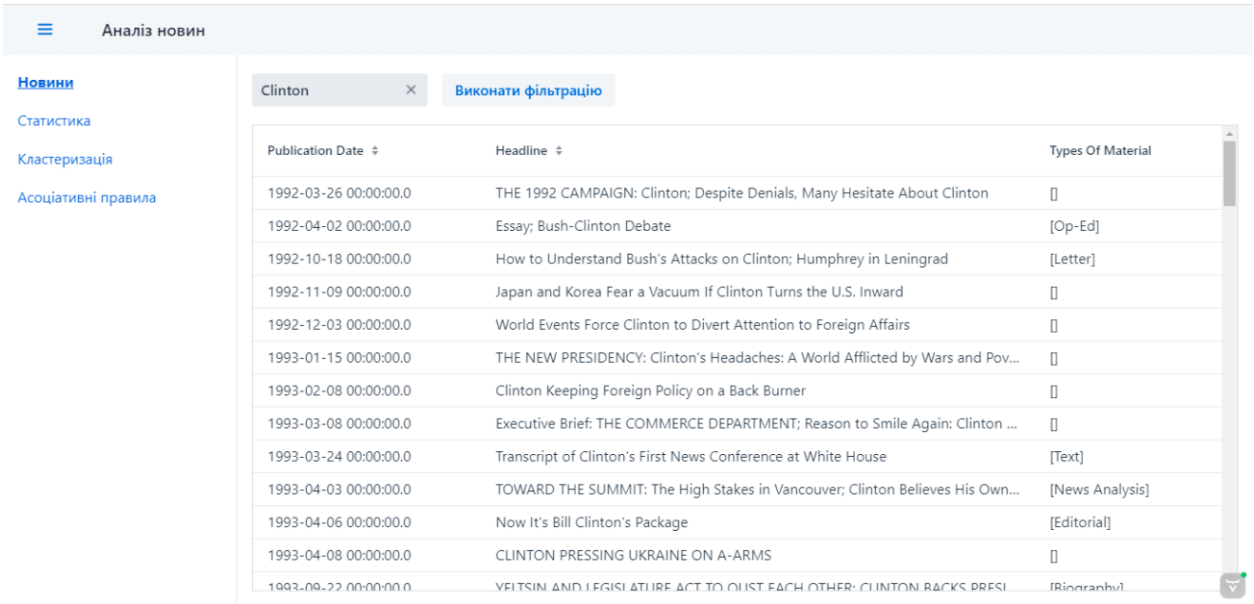

Рисунок 3.3 – Головна сторінка з новинами відфільтрованими за словом у заголовку

Натиснемо на елемент 1. Система згорне навігаційне меню, що показано на рисунку 3.4.

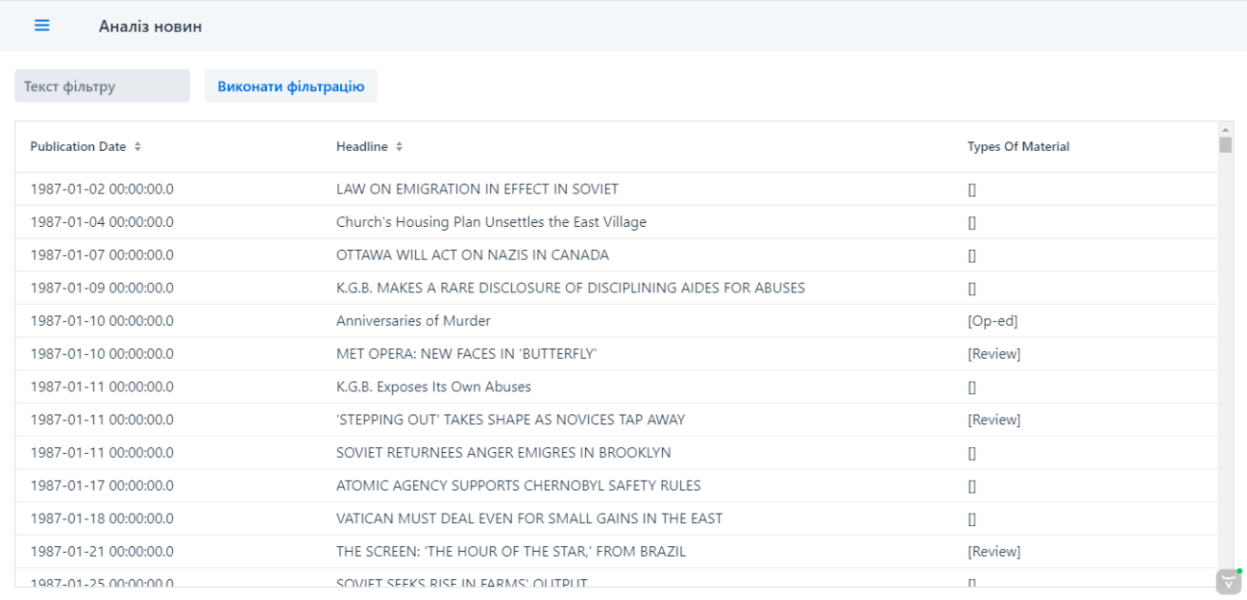

Рисунок 3.4 – Головна сторінка зі згорнутим навігаційним меню

В елементі 5, сітці новин, натиснемо на одну із новин. Виникає бічна панель із детальним зображенням новини. Це зображено на рисунку 3.5.

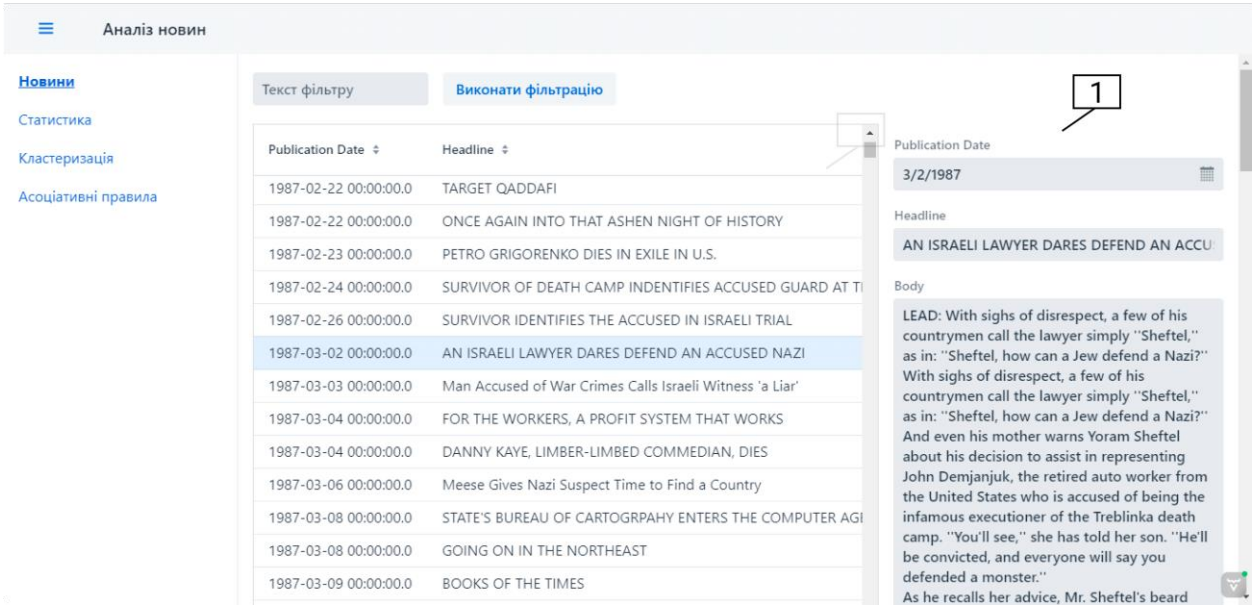

Рисунок 3.5 – Відображення повного тексту обраної статті

На рисунку 3.5 в елементі 1 зображено панель, що містить докладну інформацію про статтю, а саме дату публікації, заголовок та основне – текст статті.

Щоб перейти до статистичних відомостей натиснемо кнопку «Статистика» на елементі 6 головної сторінки. Система направить на сторінку «Статистика». Про це можна пересвідчитися на рисунку 3.6.

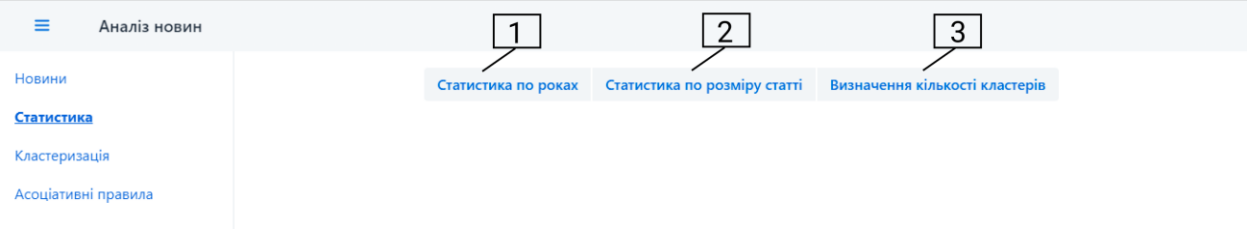

Рисунок 3.6 – Сторінка «Статистика»

Користувач має можливість переглянути три опції, а саме:

– елемент 1 – переглянути статистику статей по роках;
– елемент 2 – переглянути статистику по розміру статті (кількості слів в ній);
– елемент 3 – переглянути графік визначення кількості кластерів.

Натискаємо на елемент 1. Система відображає нам статистику кількості статей по роках. Переглянемо це на рисунку 3.7.

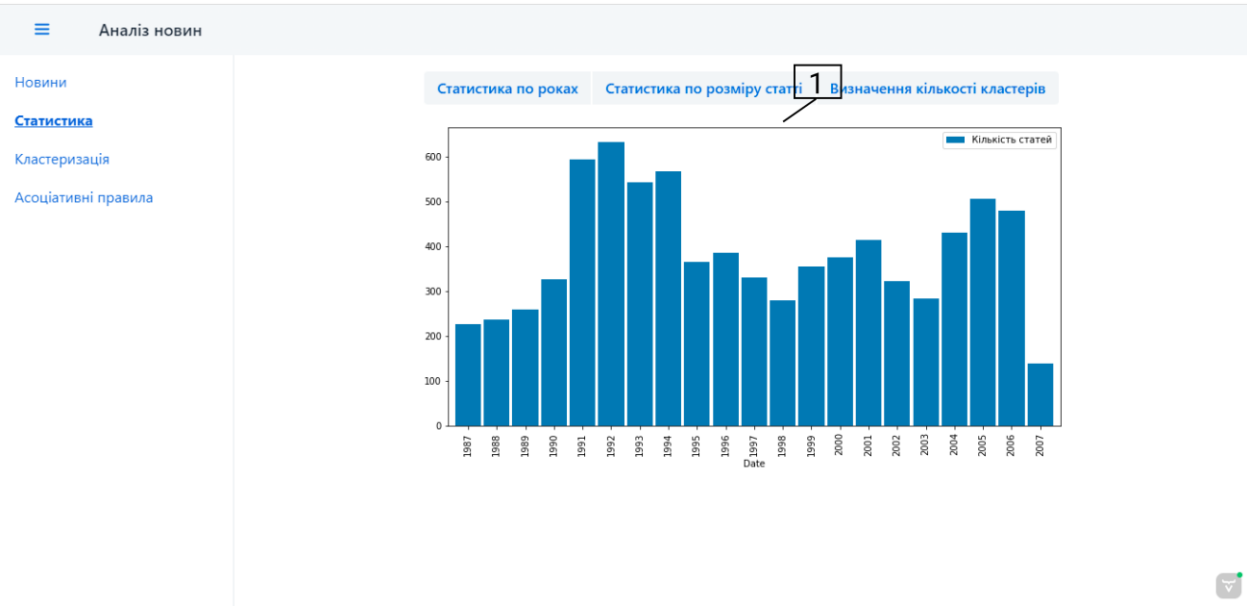

Рисунок 3.7 – Відображення кількості статей про Україну по роках

Натискаємо на елемент 2. Система показує статистику розподілу статей за кількістю слів в ній. Пересвідчитися можна на рисунку 3.8.

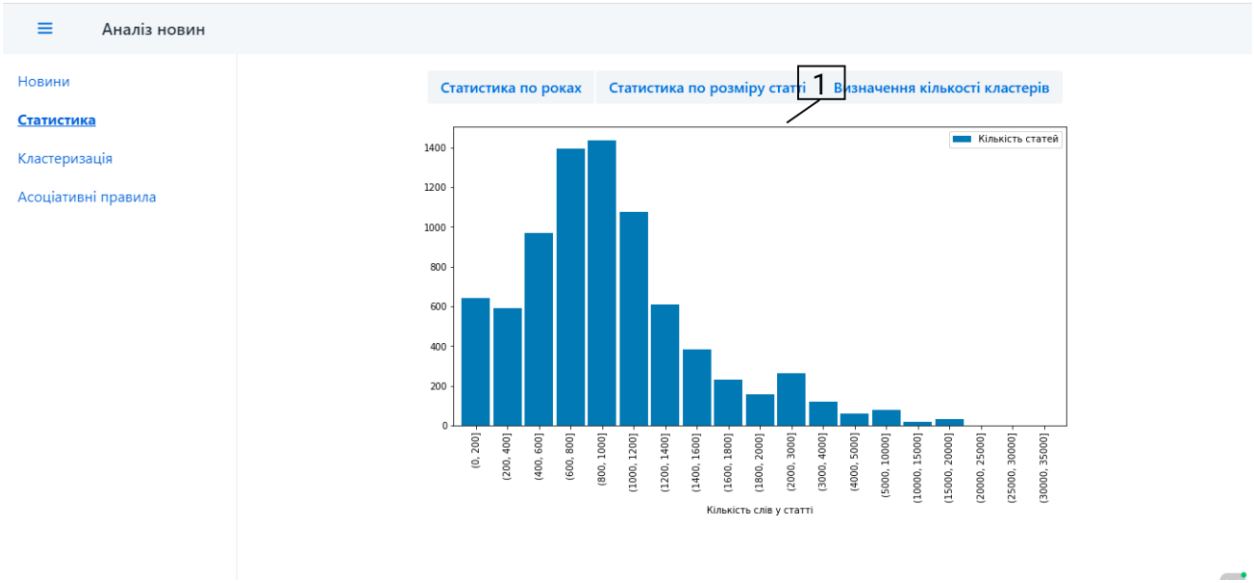

Рисунок 3.8 – Відображення кількості статей в залежності від її розміру

Щоб переглянути графік визначення кількості кластерів натискаємо на елемент 3. Система зображає графік із зображенням коефіцієнтів методу ліктя та методу силуета (рисунок 3.9).

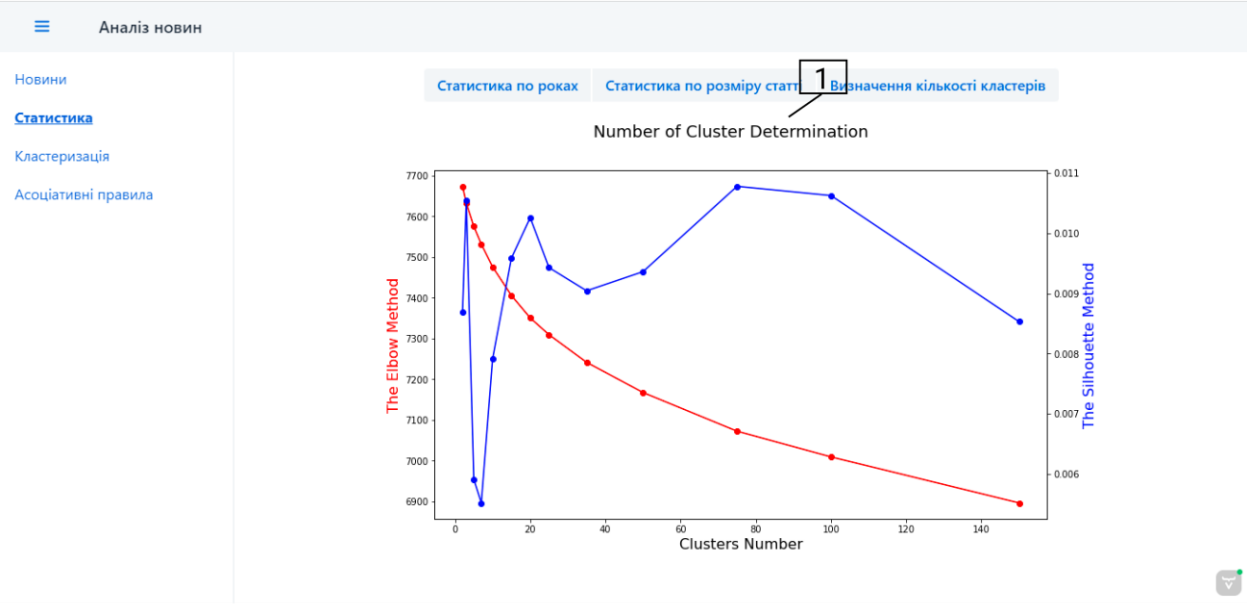

Рисунок 3.9 – Графік визначення кількості кластерів в залежності

Щоб перейти до перегляду результатів кластеризації натиснемо кнопку «Кластеризація» на елементі 6 головної сторінки. Система направляє користувача на сторінку «Кластеризація». Про це можна пересвідчитися на рисунку 3.10.

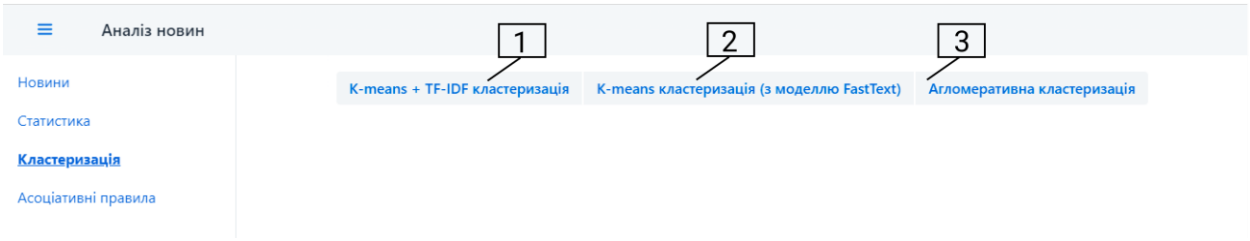

Рисунок 3.10 – Сторінка «Кластеризація»

На сторінці 3.10 можна побачити опції вибору, які результати кластеризації переглянути:

– елемент 1 – переглянути результати кластеризації методом к-середніх з використанням TF-IDF;
– елемент 2 – переглянути результати кластеризації методом к-середніх на основі матриці отриманої з використанням FastText моделі;
– елемент 3 – переглянути результати агломеративної кластеризації.

Натискаємо на елемент 3 і система відображає результати агломеративної кластеризації. Це можна переглянути на рисунку 3.11.

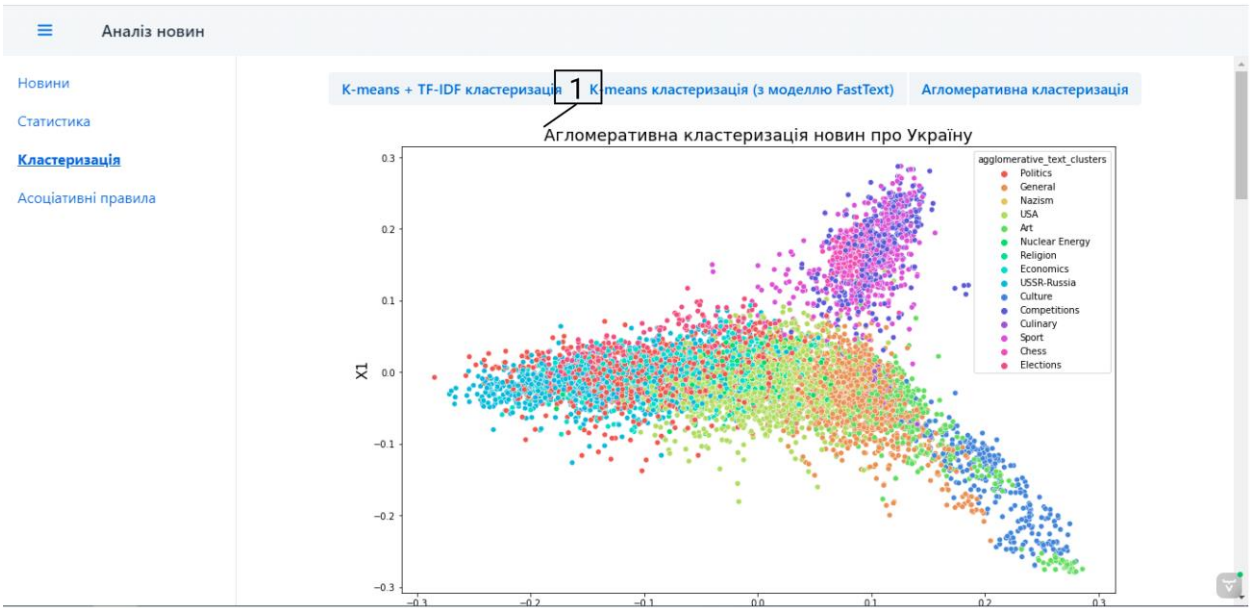

Рисунок 3.11 – Перегляд результатів кластеризації

Щоб перейти до перегляду результатів побудови асоціативних правил натискаємо кнопку «Кластеризація» в навігаційному меню. Система направляє користувача на сторінку «Асоціативні правила». Про це можна пересвідчитися на рисунку 3.12.

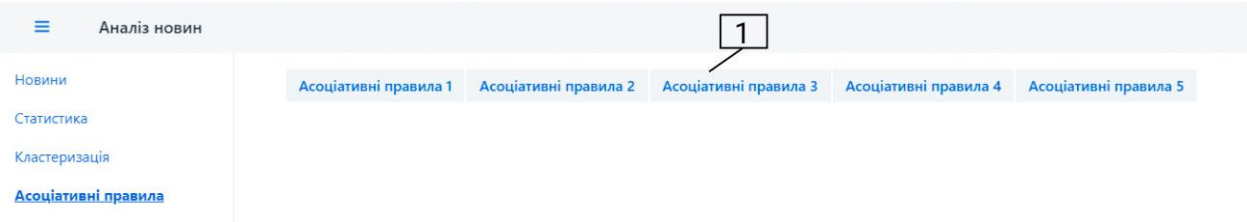

Рисунок 3.12 – Сторінка «Асоціативні правила»

В елементі 1 можна обрати асоціативні правила для певного об’єданння кластерів чи окремого кластеру. Натиснемо на один із варіантів. Система відобразить побудовані асоціативні правила у графічному вигляді, про що можна пересвідчитися на рисунку 3.13.

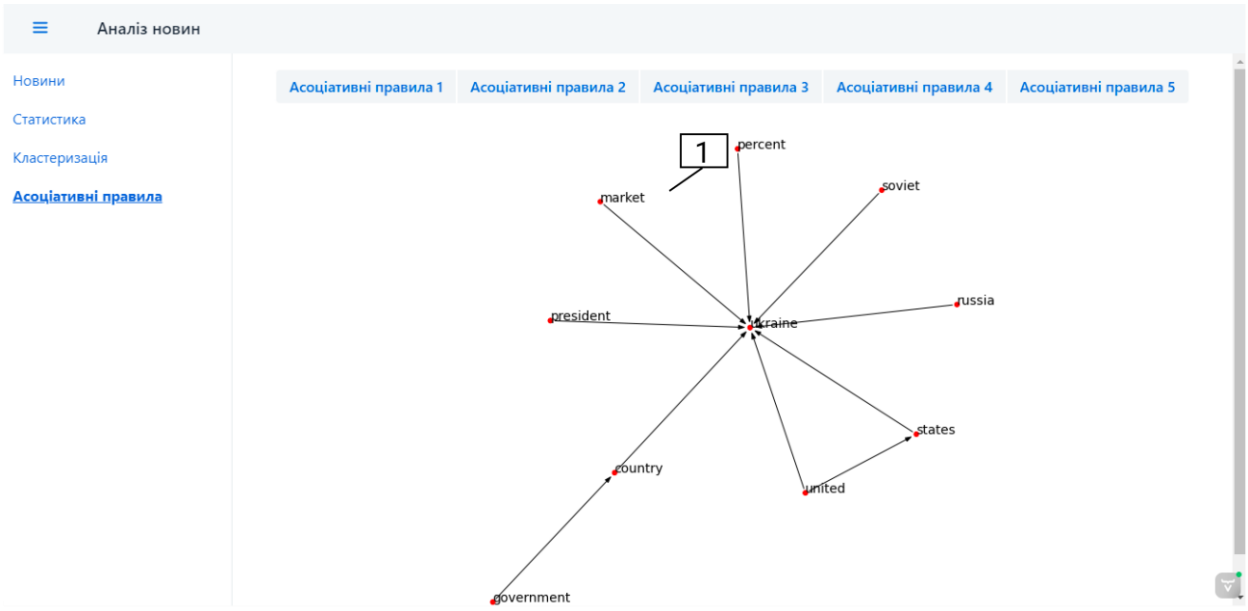

Рисунок 3.13 – Відображення побудованих асоціативних правил у вигляді графа

3.5 Висновки до розділу

Розділ 3 було присвячено розгляду програмних засобів розробки застосунку та окремі моменти розробки інформаційної системи аналізу змісту новин та прогнозування подій на його основі. Надано короткий опис використаних технологій, фремворків та бібліотеки, що їх було використано для розробки системи. Наведено вимоги до технічого забезпечення та програмних компонентів.

Застосунок розроблено із застосуванням мов програмування Java, Python, фреймовків Vaadin та Spring Boot. Розробка велася в інтерактивних середовищах IntelliJ IDEA Ultimate Student Edition та Google Colab. Використано бібліотеки Hibernate, mlxtend, scikit-learn.

Визначено та надано опис архітектури програмного забезпечення. Показано структурні схеми класів та компонентів. Детально описано основні функції програмного забезпечення.

Користувацька інструкція дає вказівки як послуговуватися систему з описом сторінок системи.

4 АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

4.1 Вхідні дані

Даними, які використовуються для проведення досліджень в рамках цієї роботи є The New York Times Annotated Corpus. Корпус даних статей New York Times, що містить понад 1,8 мільйона статей, що були написані й опубліковані з 1 січня 1987 року по 19 червня 2007 року, а також метадані статей, отримані від відділу новин видання. Загалом понад 650 тисяч статей є статтями-резюме. Понад 1,5 мільйона статей були ручну помічені тегами, отриманим зі словника індексації людей, організацій, локацій і дескрипторів тем. Понад 275 тисяч статей з автоматично проставленими тегами. Автором корпусу є Еван Сандхаус, володарем прав на всі дані корпусу є The New York Times. Користувацька ліцензія на використання даних міститься у додатку Г. Згідно з інформацією автора корпусу дані можуть використовуватися для широкого спектра задач обробки й дослідження природної мови, зокрема для розробки й оцінки алгоритмів автоматизованого узагальнення документів, виявлення сутностей в документах, категоризації документів, пошуку інформації в тексті та побудови міждокументарних посилань. Проте сфера застосування не обмежується наведеним списком. Статті корпусу написані виключно англійською мовою із дотриманням керівництва по стилю та використанню New York Times (ISBN-10 0812963881) [43].

Дані представлено у вигляді колекції XML-документів, що відповідають специфікації News Industry Text Format (NITF) версії 3.3. NITF – специфікація XML, що надає стандартизоване представлення і структуру окремих статей. NITF містить в собі розмітку, як-от підписи, заголовки, абзаци. Ще формат надає атрибути керування для категоризації статей по темах, узагальнення змін використання та історії змін. Основна мета NITF – відповісти на питання, що характерні для новин: хто, що, коли, де і чому. Хто власник авторських прав на статтю, у кого є права на

перепублікацію і головне – про кого стаття. Що – суб'єкти статті, названі в ній сутності, і події, що вона описує. Коли – час написання, публікації, редагування. Де – відповідь на питання де написана стаття, де відбувалися описані в ній події. І чому – метадані, що описують цінність статті як новини.

Пропонується роглянути приклад однієї зі статей на рисунку 4.1.

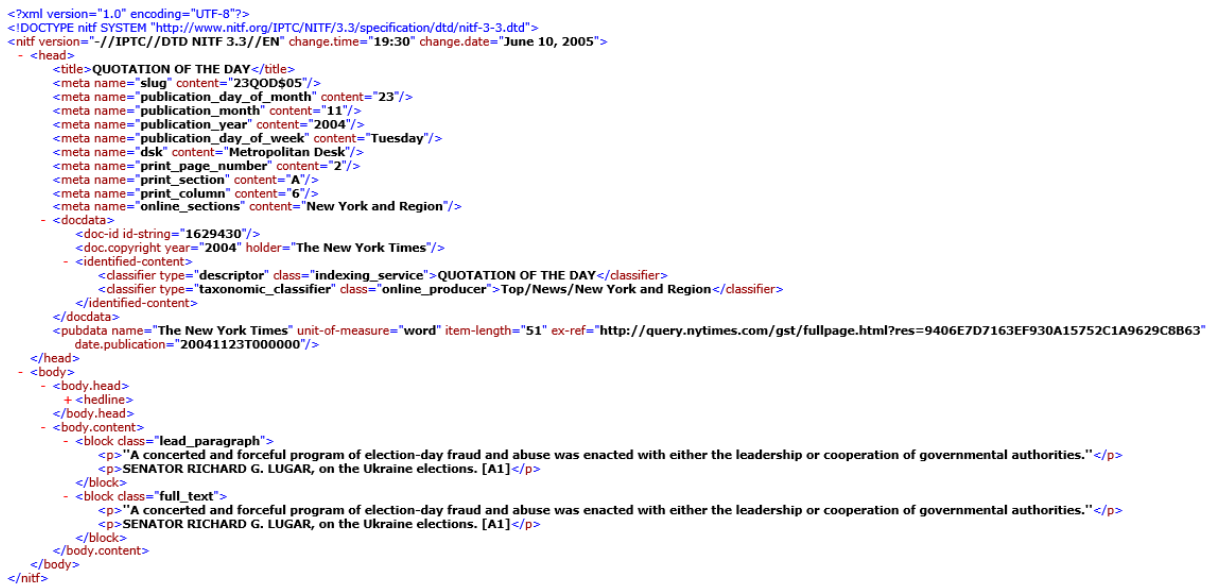

Рисунок 4.1 – Приклад вмісту xlm-файлу статті

4.2 Аналіз отриманих результатів

4.2.1 Визначення найкращої кількості кластерів

Використовуючи евклідову відстань було проведено серію експериментів на визначення найкращої кількості кластерів. Було проведено кластеризацію методом к-середніх для такої кількості кластерів $k = \{2,3,5,7,10,15,20,25,35,50,75,100,150\}$. На рисунку 4.2 показано отримані результати.

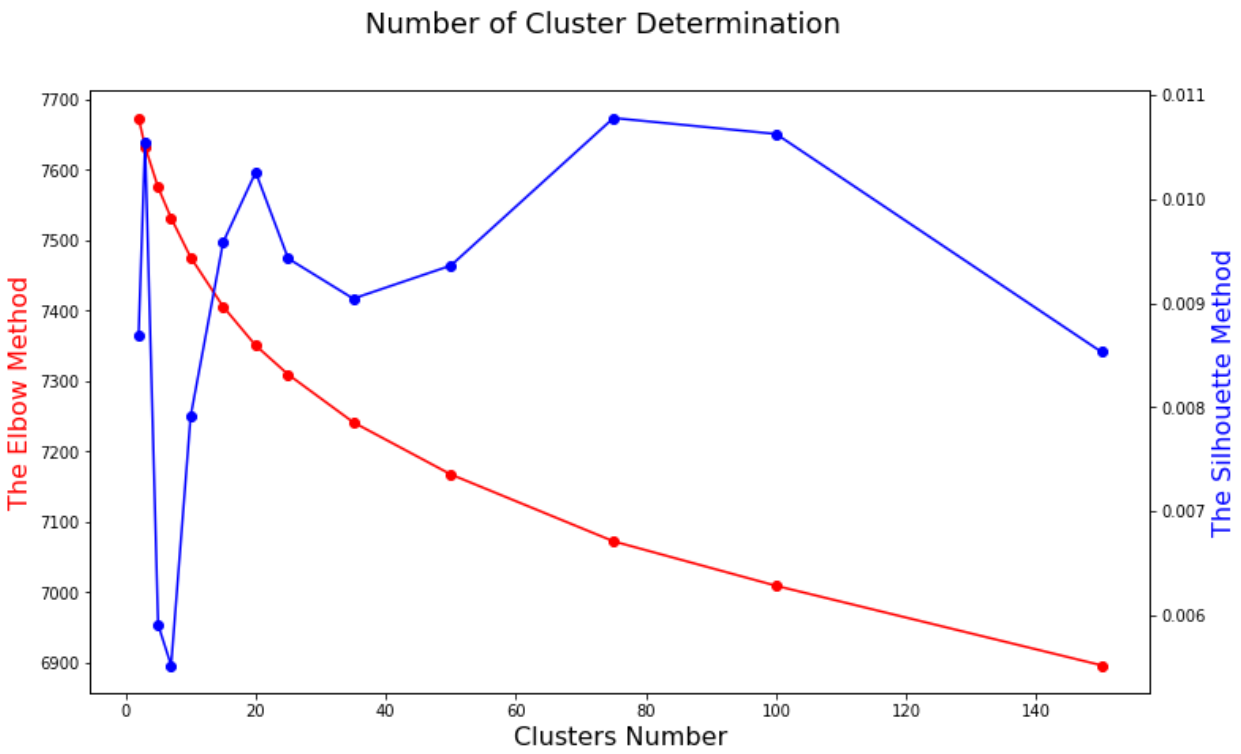

Рисунок 4.2 – Визначення найкращої кількості кластерів

На осі Ox показана кількість кластерів, для яких проводилося дослідження, а на двох осях Oy показано значення коефіцієнтів ліктя (червоним кольором) та силуету (синім кольором). Вирішено обрати $k = 15$, адже зважаючи на те, що це і так досить велика кількість семантично різних кластерів, хоч можна помітити, що значення коефіцієнта ліктя і надалі спадає, а значення коефіцієнта силуету показує максимум в точці $k = 75$.

Для того, щоб обрати метрику відстані, якою найкраще послуговувати для кластеризації обраного набору даних було поведено серію експериментів з різними метриками відстані, щоб визначити, яка найкраще підходить для даної задачі.

Обрано невелику кількість кластерів, а саме 9 і визначено значення коефіцієнтів ліктя та силуета для вказаних методів у таблиці 4.1.

Таблиця 4.1 – Дослідження коефіцієнтів ліктя і силуета для різних метрик відстаней

Метрика	Значення коефіцієнта ліктя	Значення коефіцієнта силуета
Евклідова відстань	4780	0.094015
Метрика міста	66027	0.090733
Відстань Чебишева	837	0.063683
Відстань хі-квадрат	29034	0.085552

В результаті аналізу отриманих показників визначено, що евклідова відстань найкраще підходить як метрика для визначення відстаней між об’єктами в задачі кластеризації наведених статей. Саме ця відстань показує найкраще значення для коефіцієнта силуету і друге місце для коефіцієнта ліктя, при цьому відстань Чебишева, яка показує найкращий результат методом ліктя показує найкращий результат методом силуету.

4.2.2 Представлення результатів кластеризації

Пропонується розглянути графічне представлення побудованих кластерів і об’єктів, що до них відносяться на графіках. На рисунках 4.3 – 4.5 показано побудовані кластери з використанням різних методів.

Рисунок 4.3 – Графічне представлення побудованих кластерів методом к-середніх з використанням TF-IDF

Рисунок 4.4 – Графічне представлення побудованих кластерів методом к-середніх з використанням TF-IDF

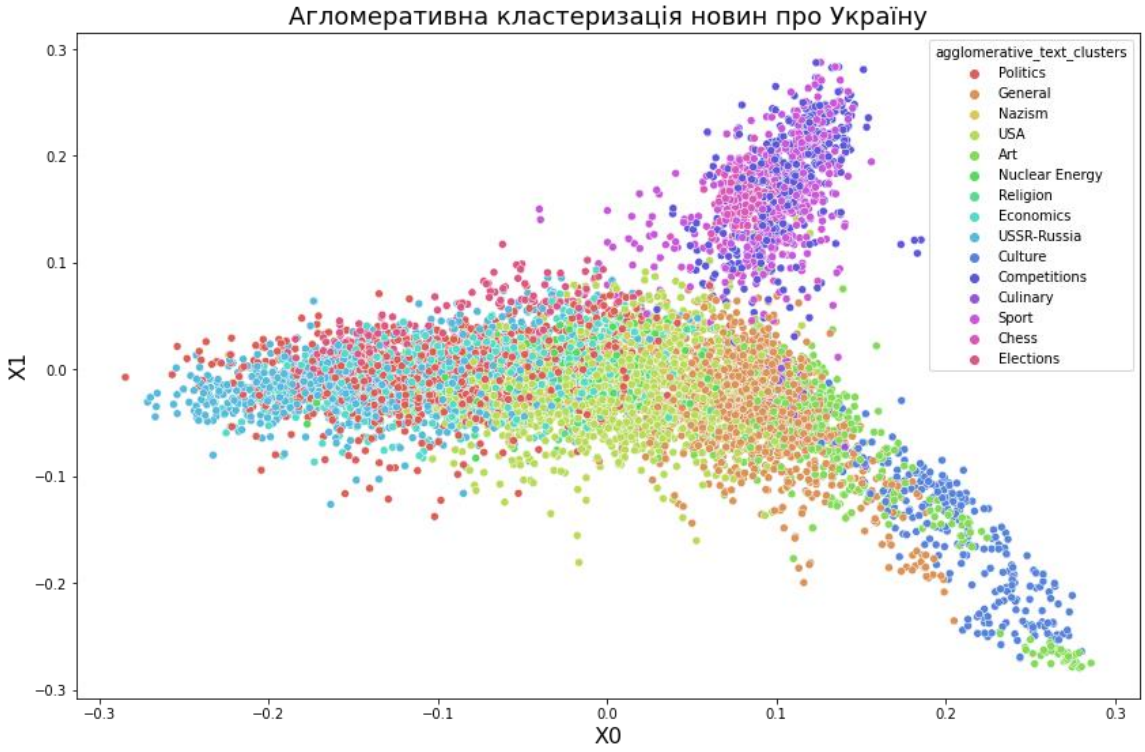

Рисунок 4.5 – Графічне представлення побудованих кластерів агломеративною кластеризацією

Як видно з рисунків 4.3 – 4.5 побудовані трьома різними способами кластери не є ідентичними, хоч і часто збігаються. Це свідчить про те, що складно ідеально розбити на кластери набір текстових даних, однак закономірності видимі та очевидні.

На рисунках 4.6 – 4.8 подано матриці з коефіцієнтами Жаккара для трьох виконаних кластеризацій.

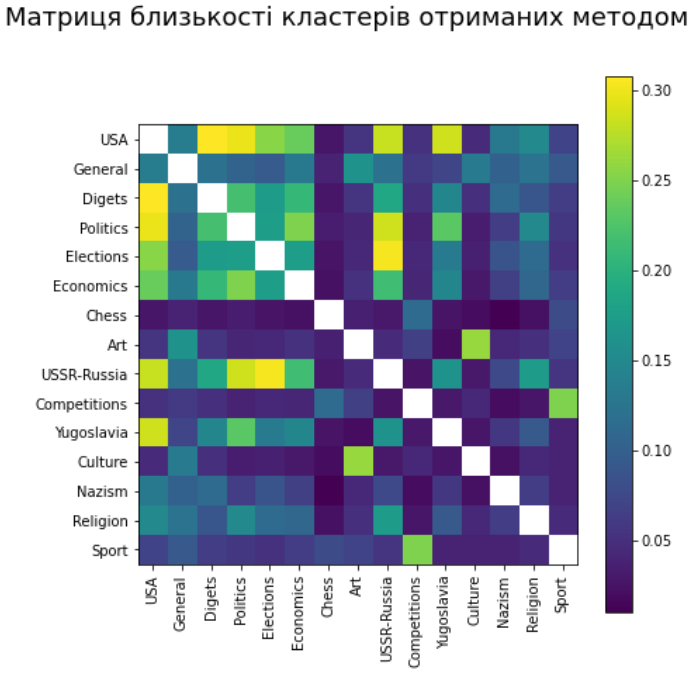

Рисунок 4.6 – Матриця подібності (близькості) кластерів отриманих методом к-середніх

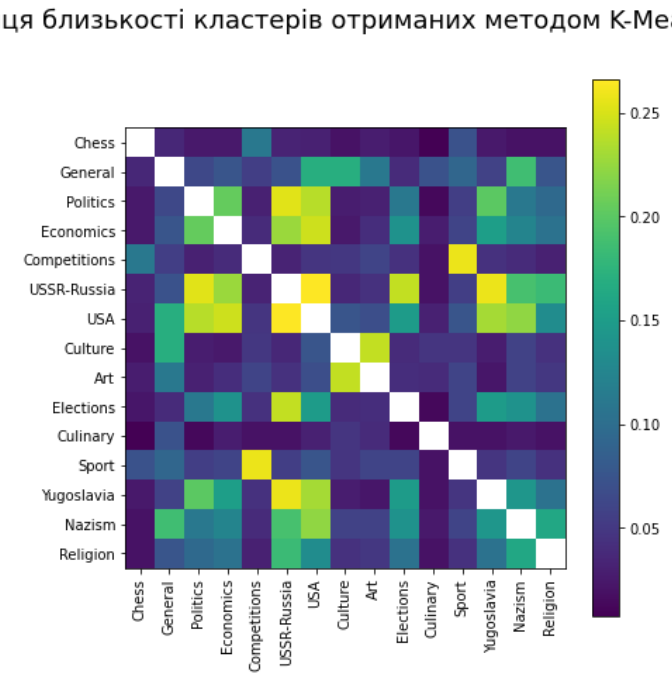

Рисунок 4.7 – Матриця подібності (близькості) кластерів отриманих методом к-середніх з використанням FastText + word2vec

Рисунок 4.8 – Матриця подібності (близькості) кластерів отриманих в результаті агломеративної кластеризації

На рисунках 4.6 – 4.8 помітно, що найчастіше коефіцієнт Жаккара для будь-яких двох кластерів зовсім невеликий, однак, для деяких кластерів він суттєво більший, що показує тематичну і змістовну близькість статей, що належать до даних кластерів. Можна навести декілька таких прикладів, як от певна близькість кластера 'USA' до кластерів ‘Yugoslavia’ і ‘USSR-Russia’. Іншим яскравим прикладом є близькість кластерів 'Art' і 'Culture' чи близькість між собою кластерів 'Sport', 'Competitions' і 'Chess', де вони всі три виділяються. Останні твердження є абсолютно недивними, оскільки різниця між мистецтвом і культурою направду невелика, як і те, що змагання часто є складовою спорту, а шахи, хоч і особливий, але вид спорту.

З метою найкращої візуалізації ключових слів виконано побудову хмар слів для усіх кластерів кожного з трьох варіантів кластеризації. Результати даної побудови для кластеризації методом к-середніх можна побачити на рисунках 4.9 – 4.10, а для інших методів у додатку Б.

Рисунок 4.9 – Ключові слова по кластерах (метод к-середніх). Частина 1

Рисунок 4.10 – Ключові слова по кластерах (метод к-середніх). Частина 2

На наведених вище рисунках 4.9 і 4.10 можна пересвідчитися, що розбиття на кластери досить якісне. Наприклад, розглянемо ключові слова кластера 'Competitions'. На хмарі слів можна побачити такі слова, як: player, game, team, victory, round, champion, beat. Вони дійсно характеризують даний кластер і відрізняються від ключових слів інших кластерів.

4.2.3 Представлення результатів побудови асоціативних правил

На рисунках 4.11 – 4.13 переглянемо результати побудови асоціативних правил.

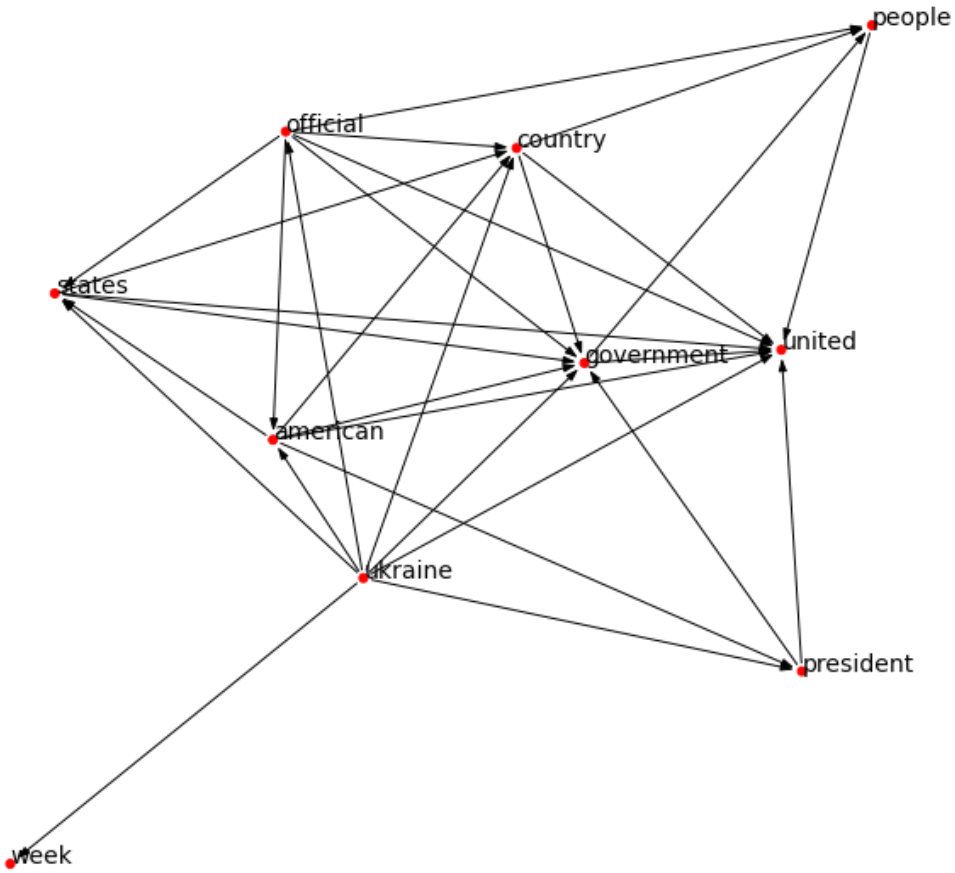

Рисунок 4.11 – Побудовані асоціативні правила для кластерів ‘USA’ та ‘Digets’

З рисунка 4.11 можна побачити такі причинно-наслідоків ланцюги, як, наприклад: ukraine → american → states → country → government → people → united чи ukraine → american → states → united. Семантично знайдені ланцюги правил можна пояснити так: новина, яка містить слово «Україна» найчастіше міститиме в тексті новини згадки про зв’язки зі США, урядом, впливом на людей і сполученим, що може бути, як в контексті США, так і в контексті Сполученого Королівства Великої Британії. Зважаючи на те, що дані опрацьовано за період з 1987 по 2007 роки, а ситуація станом на сьогодні відповідає цьому прогнозу: найближчими партнерами України є США та Велика Британія, уряди яких надають Україні обширну підтримку та допомогу.

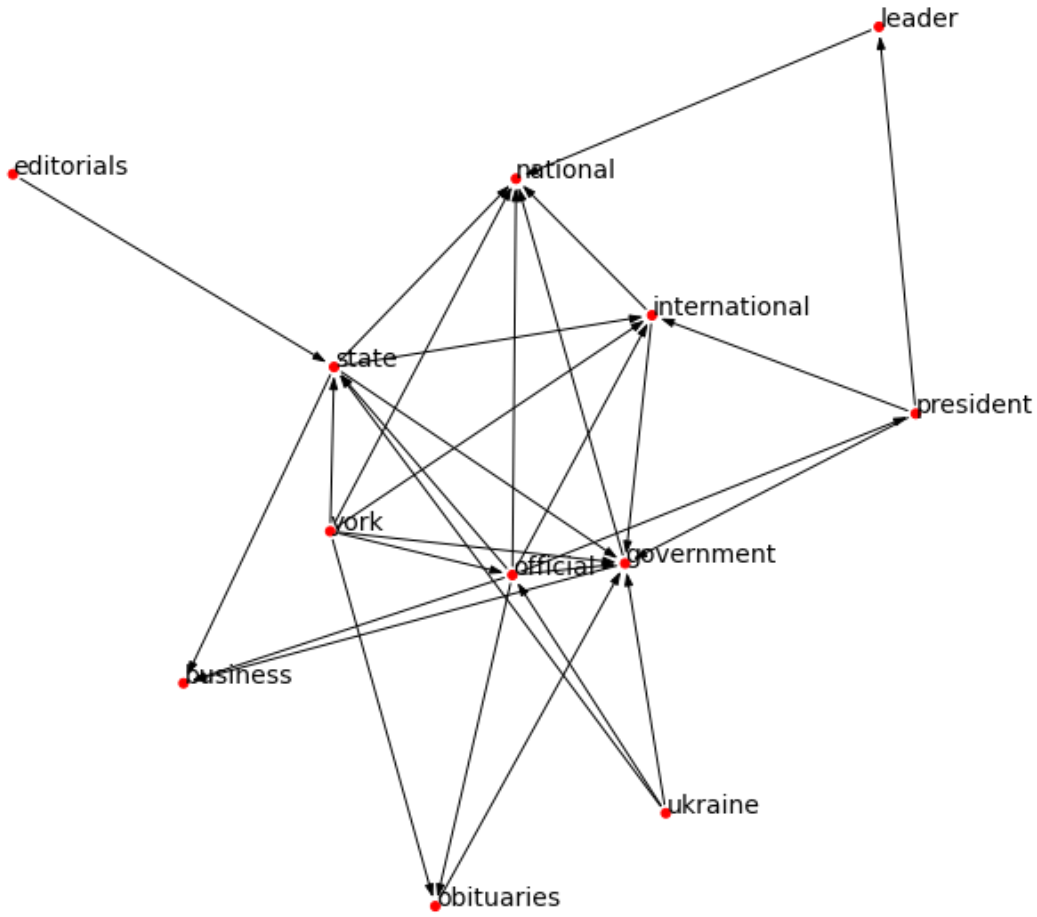

Рисунок 4.12 – Побудовані асоціативні правила для кластерів ‘USA’ та ‘Politics’

З рисунка 4.12 можна побачити такі правила: york → official → president → international → government → business чи editorials → state → international → government → national. Семантика першого зрозуміла. Саме місто Нью-Йорк є центром світвої політики, і президент США офіційно керує міжнародними відносинами країни, офіціною політикою уряду в тому числі по відношенню до різного бізнесу, включаючи міжнародний. Другий ланцюг теж зрозумілий. Редакційні статті (editorials) є саме такими масштабними, щоб покривати взаємодію на міжнародному та національому рівнях, зокрема роботу уряду країни. Більше того, США залишається лідером на міжнародній політичній та економічній арені й

сьогодні, а президент США – найвпливовіший світовий лідер, що підтверджує прогнозування.

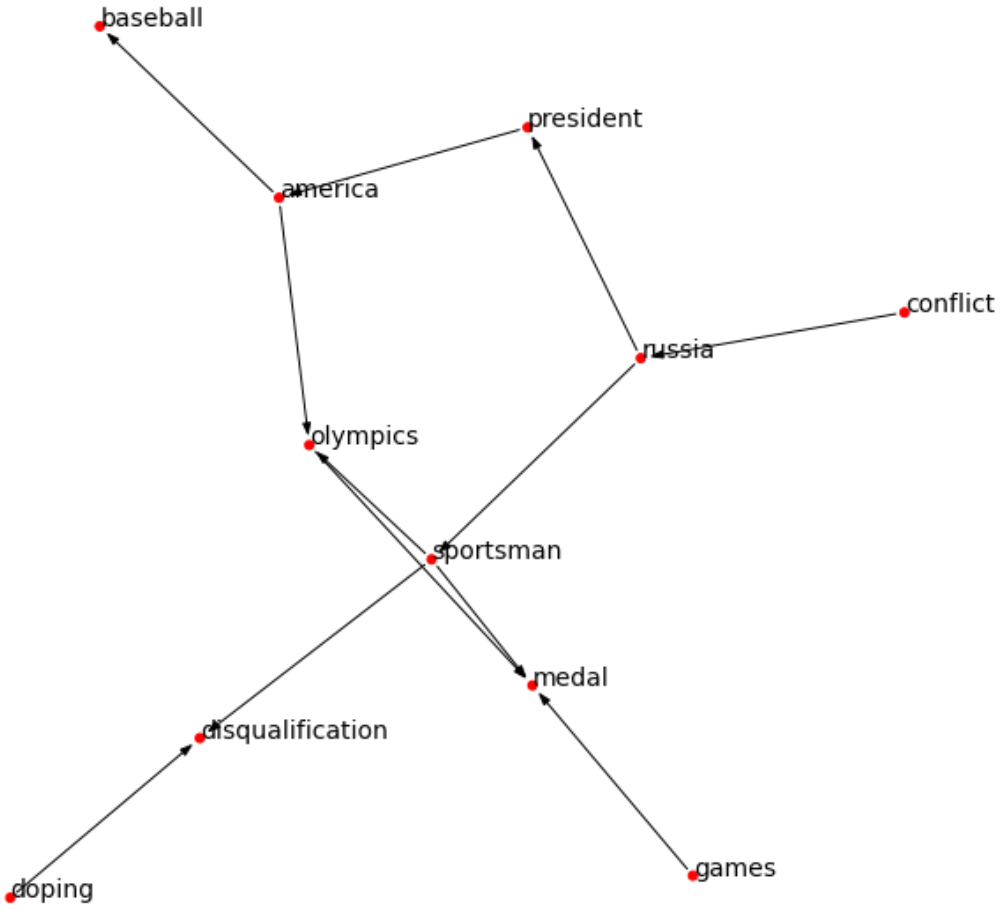

Рисунок 4.13 – Побудовані асоціативні правила для кластерів ‘Politics’ та ‘Sport’

На рисунку 4.13 особливо цікавим є правило conflict → russia → sportsman → disqualification. Його варто розуміти так, що у великій кількості конфліктів часто бере участь Росія. Вона ж часто поширює і популяризує себе шляхом висвітлення досягнень своїх спортсменів. Атлетів же часто дискваліфіковують за різні порушення (ланцюг doping → disqualification також присутній). Себто, з цього ланцюга можна отримати прогноз, що російські спортсмени будуть дискваліфіковані у зв’язку з певним конфліком. Це і трапилося зараз у зв’язку з військовим вторгненням РФ на територію України.

Варто зауважити, що вхідний набір даних охоплює новини з 1987 до 2007 року, а ланцюги актуальні й зараз. Фактично в якості прогнозу отримано опис події (без вказівки дати настання), яка сталася в результаті аналізу подій з минулого.

4.3 Висновки до розділу

У цьому розділі описано вхідні дані, що використовуються при виконанні досліжень. Показано результати експериментального визначення кількості кластерів та вибору метрики для визначення відстані між об’єктами. У графічному представлені показано близькість (подібність) кластерів, а також надано хмари слів для кластерів.

Наведено підхід до побудови ланцюга асоціативних правил з метою прогнозування подій. Показано приклади виявлених ланцюгів асоціативних правил та пояснено їх семантичне значення. Хоч використана новинна база і не найсучасніша (1987-2007 роки), але отримані ланцюги асоціативних правил є актуальними й нині.

ВИСНОВКИ

Магістерська робота присвячена темі аналізу змісту новин та прогнозування подій на його основі. Актульність теми обумовлена зростанням кількості новин та виявленням впливу новин на події, що відбуваються у світі. В першу чергу було проведено огляд відомих підходів та методів розв’язання задачі, в межах огляду визначено переваги та особливості наявних методів. Окрема увага приділена питанню якості новин, вибору джерела вхідних даних, визначено критерії якості та обрано джерело.

Послуговуючись оглядом сучасних досліджень з теми дисертації було розроблено узагальнений підхід до аналізу змісту новин та прогнозування подій на його основі. Застосування кластеризації текстових даних дозволило розбити новини на спільні групи (кластери) та об’єднати згодом найбільш подібні для побудови асоціативних правил.

Під час розробки підходу асоціативних правил було запропоновано удосконалити метод для вирішення задачі побудови прогнозів, шляхом утворення ланцюгів правил (саме ці ланцюги і є прогнозами).

На основі розробленого алгоритмічного забезпечення було розроблено прототип інформаційної системи, що дозволяє виконувати аналіз новин та прогнозування подій. Розроблено керівництво користувача, яке надає інструкцію щодо того як послуговуватися системою.

За матеріалами дослідження було опубліковано 2 наукових роботи: 1 стаття у міжвідомчому науково-технічному збірнику «Адаптивні системи автоматичного управління» та тези на першій всеукраїнській науково-практичній конференції молодих вчених та студентів «Інженерія програмного забезпечення і передові інформаційні технології» (SoftTech-2021).

ПЕРЕЛІК ПОСИЛАНЬ

1. Процюк Ю. В., Гавриленко О. В. Approaches to the solution to the problem of news-based events forecasting. Адаптивні системи автоматичного управління. 2022. Т. 1, № 40. С. 15–20.
2. Процюк Ю. В., Гавриленко О. В. Огляд задачі прогнозування подій на основі новин та методів її розв'язання. Перша Всеукраїнська науково-практична конференція молодих вчених та студентів «Інженерія програмного забезпечення і передові інформаційні технології» (SoftTech-2021) : Матеріали всеукраїнської науково-практичної конференції молодих вчених та студентів, м. Київ, 22 листоп. 2021 р. – 26 трав. 2022 р. Київ, 2021. С. 150–153.
3. Computational Linguistics (Stanford Encyclopedia of Philosophy). Stanford Encyclopedia of Philosophy. URL: https://plato.stanford.edu/entries/computational-linguistics/ (date of access: 24.05.2022).
4. Ganesan K. All you need to know about text preprocessing for NLP and Machine Learning - KDnuggets. KDnuggets. URL: https://www.kdnuggets.com/2019/04/text-preprocessing-nlp-machine-learning.html (date of access: 24.05.2022).
5. Gavrilenko O., Oliinyk Y., Khanko H. Analysis of Propaganda Elements Detecting Algorithms in Text Data. Advances in Computer Science for Engineering and Education II : International Conference, 29 March 2019. 2019. P. 438–447. URL: https://doi.org/10.1007/978-3-030-16621-2_41 (date of access: 24.05.2022).
6. Boehmke B., Greenwell B. Chapter 21 Hierarchical Clustering | Hands-On Machine Learning with R. GitHub Pages. URL: https://bradleyboehmke.github.io/HOML/hierarchical.html#hierarchical-clustering-algorithms (date of access: 24.05.2022).

7. Jain A. K. Algorithms for clustering data. Englewood Cliffs, N.J : Prentice Hall, 1988. 320 p.
8. Kaufman L., Rousseeuw P. J. Finding Groups in Data: An Introduction to Cluster Analysis. Wiley & Sons, Incorporated, John, 2009. 342 p.
9. Spatio-temporal Event Forecasting and Precursor Identification / Y. Ning et al. KDD '19: The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Anchorage AK USA. New York, NY, USA, 2019. URL: https://doi.org/10.1145/3292500.3332291 (date of access: 24.05.2022).
10. Association rule mining / ed. by C. Zhang, S. Zhang. Berlin, Heidelberg : Springer Berlin Heidelberg, 2002. URL: https://doi.org/10.1007/3-540-46027-6 (date of access: 24.05.2022).
11. Political Information Opportunities in Europe / F. Esser et al. The International Journal of Press/Politics. 2012. Vol. 17, no. 3. P. 247–274. URL: https://doi.org/10.1177/1940161212442956 (date of access: 24.05.2022).
12. Hjarvard S., Kammer A. Online news: between private enterprise and public subsidy. Media, Culture & Society. 2014. Vol. 37, no. 1. P. 115–123. URL: https://doi.org/10.1177/0163443714553562 (date of access: 24.05.2022).
13. Building Empirical Typologies with QCA: Toward a Classification of Media Systems / F. Büchel et al. The International Journal of Press/Politics. 2016. Vol. 21, no. 2. P. 209–232. URL: https://doi.org/10.1177/1940161215626567 (date of access: 25.05.2022).
14. Peifer J. T. Perceived News Media Importance: Developing and Validating a Measure for Personal Valuations of Normative Journalistic Functions. Communication Methods and Measures. 2018. Vol. 12, no. 1. P. 55–79. URL: https://doi.org/10.1080/19312458.2017.1416342 (date of access: 24.05.2022).
15. Hanitzsch T., Van Dalen A., Steindl N. Caught in the Nexus: A Comparative and Longitudinal Analysis of Public Trust in the Press. The International Journal of

Press/Politics. 2017. Vol. 23, no. 1. P. 3–23. URL: https://doi.org/10.1177/1940161217740695 (date of access: 24.05.2022).

16. Bachmann P., Eisenegger M., Ingenhoff D. Defining and Measuring News Media Quality: Comparing the Content Perspective and the Audience Perspective. The International Journal of Press/Politics. 2021. P. 9–37. URL: https://doi.org/10.1177/1940161221999666 (date of access: 24.05.2022).
17. Glader P. 10 Journalism Brands Where You Find Real Facts Rather Than Alternative Facts. Forbes. URL: https://www.forbes.com/sites/berlinschoolofcreativeleadership/2017/02/01/10-journalism-brands-where-you-will-find-real-facts-rather-than-alternative-facts/ (date of access: 24.05.2022).
18. Discovering and learning sensational episodes of news events / X. Ao et al. Information Systems. 2018. Vol. 78. P. 68–80. URL: https://doi.org/10.1016/j.is.2018.05.003 (date of access: 24.05.2022).
19. Preethi P. G., Uma V., kumar A. Temporal Sentiment Analysis and Causal Rules Extraction from Tweets for Event Prediction. Procedia Computer Science. 2015. Vol. 48. P. 84–89. URL: https://doi.org/10.1016/j.procs.2015.04.154 (date of access: 24.05.2022).
20. Gerber M. S. Predicting crime using Twitter and kernel density estimation. Decision Support Systems. 2014. Vol. 61. P. 115–125. URL: https://doi.org/10.1016/j.dss.2014.02.003 (date of access: 24.05.2022).
21. Realization of a news dissemination agent based on weighted association rules and text mining techniques / C.-J. Huang et al. Expert Systems with Applications. 2010. Vol. 37, no. 9. P. 6409–6413. URL: https://doi.org/10.1016/j.eswa.2010.02.078 (date of access: 24.05.2022).
22. Modeling Precursors for Event Forecasting via Nested Multi-Instance Learning / Y. Ning et al. KDD '16: The 22nd ACM SIGKDD International Conference on

Knowledge Discovery and Data Mining, San Francisco California USA. New York, NY, USA, 2016. URL: https://doi.org/10.1145/2939672.2939802 (date of access: 24.05.2022).

23. Singh S., Khatri R. Data Mining based Technique for Natural Event Prediction and Disaster Management. International Journal of Computer Applications. 2016. Vol. 139, no. 14. P. 34–39. URL: https://doi.org/10.5120/ijca2016909102 (date of access: 24.05.2022).
24. What is Text Mining, Text Analytics and Natural Language Processing? Linguamatics. NLP Text Mining Products and Solutions for Healthcare and Pharma | Linguamatics. URL: https://www.linguamatics.com/what-text-mining-text-analytics-and-natural-language-processing (date of access: 24.05.2022).
25. Ramadhan L. TF-IDF Simplified. Towards Data Science. URL: https://towardsdatascience.com/tf-idf-simplified-aba19d5f5530 (date of access: 24.05.2022).
26. Методы кластеризации. Кафедра програмного забезпечення Дніпровського державного технічного університету. URL: http://pzs.dstu.dp.ua/DataMining/cluster/index.html (дата звернення: 24.05.2022).
27. Malik U. Hierarchical Clustering with Python and Scikit-Learn. Stack Abuse. URL: https://stackabuse.com/hierarchical-clustering-with-python-and-scikit-learn/ (date of access: 24.05.2022).
28. Mahendru K. How to Determine the Optimal K for K-Means?. Medium. URL: https://medium.com/analytics-vidhya/how-to-determine-the-optimal-k-for-k-means-708505d204eb (date of access: 25.05.2022).
29. Sklearn.metrics.pairwise_distances. scikit-learn. URL: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise_distances.html (date of access: 24.05.2022).

30. Гавриленко О. В. Навчальний посібник з дисциплін “Аналіз даних” та “Аналіз даних в управляючих системах” для студентів спеціальності 126 : навчальний посібник. Київ : НТУУ «КПІ ім. Ігоря Сікорського», 2020.
31. Jaadi Z. A Step-by-Step Explanation of Principal Component Analysis (PCA). Built In. URL: https://builtin.com/data-science/step-step-explanation-principal-component-analysis (date of access: 24.05.2022).
32. Sayad S. Association Rules. Data Mining Map. URL: https://www.saedsayad.com/association_rules.htm (date of access: 24.05.2022).
33. Frequent Pattern (FP) Growth Algorithm In Data Mining. Software Testing Help. URL: https://www.softwaretestinghelp.com/fp-growth-algorithm-data-mining/ (date of access: 24.05.2022).
34. Top 9 Practical Benefits of Java Web Applications. Nexus Web Development Company. URL: https://nexwebsites.com/website-development/top-9-practical-benefits-of-java-web-applications/ (date of access: 24.05.2022).
35. Spring Boot Reviews, Competitors and Pricing. Buying Intelligence and Reviews for Enterprise Technology | PeerSpot. URL: https://www.peerspot.com/products/spring-boot-reviews (date of access: 24.05.2022).
36. Waseem M. What is Hibernate in Java | Introduction to Hibernate Framework | Edureka. Edureka. URL: https://www.edureka.co/blog/what-is-hibernate-in-java/ (date of access: 24.05.2022).
37. Sharma R. 7 Advantages of using Python for Data Science | upGrad blog. upGrad blog. URL: https://www.upgrad.com/blog/advantages-of-using-python-for-data-science/ (date of access: 24.05.2022).
38. Rane Z. 10 Compelling Reasons to Learn Python for Data Science. Towards Data Science. URL: https://towardsdatascience.com/10-compelling-reasons-to-learn-python-for-data-science-fa31160321cb (date of access: 24.05.2022).

39. Techopedia. What is the Natural Language Toolkit (NLTK)? - Definition from Techopedia. Techopedia.com. 2014. URL: https://www.techopedia.com/definition/30343/natural-language-toolkit-nltk (date of access: 24.05.2022).
40. Gensim - Introduction. Online Tutorials Library. URL: https://www.tutorialspoint.com/gensim/gensim_introduction.htm (date of access: 24.05.2022).
41. Alizadeh E. MLxtend: A Python Library with Interesting Tools for Data Science Tasks. MLxtend: A Python Library with Interesting Tools for Data Science Tasks. URL: https://ealizadeh.com/blog/mlxtend-library-for-data-science (date of access: 24.05.2022).
42. Overview | Vaadin Architecture | Framework | Vaadin Docs. Vaadin - An open platform for building web apps in Java. URL: https://vaadin.com/docs/v8/framework/architecture/architecture-overview (date of access: 24.05.2022).
43. The New York Times Annotated Corpus - Linguistic Data Consortium. Linguistic Data Consortium - Linguistic Data Consortium. URL: https://catalog.ldc.upenn.edu/LDC2008T19 (date of access: 24.05.2022).

ДОДАТКИ

Додаток А Перелік опублікованих матеріалів

UDC 681.513.7; 519.688

Y. Protsiuk, O. Gavrylenko

APPROACHES TO THE SOLUTION TO THE PROBLEM OF NEWS-BASED EVENTS FORECASTING

Abstract: An overview of the areas of application of approaches and methods of forecasting events based on past events. The substantiation of urgency of a theme is given and possibilities concerning application of results of work are resulted. Requirements for incoming news regarding their quality are defined. It is noted that there are four key criteria for the quality of the media, which are often two-component, namely: the relevance of news, providing the context in which the event, compliance with professional standards and a variety of materials. The key stages of working with data in order to obtain knowledge from them for forecasting events are identified. These include pre-processing of data (reduction to a standardized view that will understand and be able to process the algorithm), their analysis and the forecasting process itself. The spheres of application of associative series and Markov processes for search of causal relations, and time series for definition of the period of occurrence of an event with the set probability are specified.

Keywords: event forecasting, data mining, news quality, computer linguistics, time series, associative rules.

Introduction

The media have long ceased to be just a source of news about world events. It is impossible to deny how much the media influences the human mind and the course of events in general. Increasingly, you can see how the news becomes a harbinger of certain events, as if calling them. Therefore, there is a question of building a model that will help predict future events based on information about the past. Gaining knowledge of natural languages is an important and little-studied issue of data mining. Creating quality models can solve the global problem of finding hidden patterns that can predict possible future news events and create models of influence on various economic and social processes. An example of predicted phenomena may be the emergence of economic crisis or social processes.

Some knowledge of the future will allow people or especially the leadership to take

Some knowledge of the future will allow people or especially the leadership to take measures to mitigate or avoid adverse events or, conversely, to create certain trends in the economy. All this can potentially affect the fate of mankind. In any case, the identified patterns are a field for scientists from different fields, not just computer scientists or sociologists. The analysis of texts and the extraction of data from them for use in forecasting models can be based on known methods of computational linguistics and text mining.

Analysis of recent publications

In general, the analysis of news to predict future events is a topic that has been little studied due to its complexity. In the study [1] with the help of natural language processing methods studied the peculiarities of the emergence of interconnected sensational news based on text analysis. The

---

article examines the patterns of occurrence of pairs of events that occurred in the news space and how to predict that the second event will occur after the first. Computational linguistics methods have been used to find cause-and-effect relationships between events from their test descriptions.

Research [2] is devoted to the identification of causal links between events on social networks to predict the tone of the event and the time between the occurrence of different events. First, messages are selected over a period of time, from which the keywords used to determine the tone of the message - positive, negative or neutral - are selected. To determine the tone of words, a classifier is used, which is studied on the basis of the method of reference vectors. In the future, causal relationships are built between keywords, using the method of associative rules, which creates rules of the form "if" from the data. The final step is to predict events using temporal analysis of messages and the calculation of causation.

Research [3] is based on linguistic analysis and statistical modeling of tweets to automatically identify topics discussed in large cities. To single out topics, it is recommended to use thematic modeling, for which the text of the tweets was divided into special tokens using a language tokenizer and a partial tag. In this case, emoticons were considered as separate tokens that carry a certain content load. To model the topic, semantic content is also analyzed, which describes the emotional state of the author of the tweet.

The problem of the influence of news headlines on the behavior of investors and changes in the financial market is covered in [4]. A model based on prudent associative rules determines whether news is important enough for investors. When learning from real-world data, the weighted associative rules algorithm finds terms that appear frequently and simultaneously in news headlines. The term appears in the headlines several times a day for a certain number of days. And the severity of the impact of the term is determined by how much the share price has changed over the period, taking into account the frequency of use of this term. These scales allow you to determine whether certain terms affect the results of trade.

Some researches propose methods to solve the problem of identifying events that are a harbinger of future events and identifies future events. According to the collection of streaming news from open sources, an embedded approach has been developed to predict significant public events and protests. The strengths of this approach are proved by empirical assessment, which consists in filtering potential precursors in order to qualitatively predict the signs of events of joint riots and in predicting the instance of an event ahead of time [5].

The authors of the study [6] present a model for predicting fatal accidents and natural disasters. The authors have collected text messages from Google about disasters. The resulting text documents were processed using computational linguistics methods and erroneous results were eliminated using a trained naive Bayesian classifier. After data collection, semantic clustering of this data was performed. The transition matrix was built from the keywords used in data collection. The observation matrix was built from grouped events. Both matrices were fed to the input of a hidden Markov model for prediction. To predict a new event with the specified topic, it is necessary to develop a model of origin based on its time series, and then find the density function of the

distribution of its parameters. When forecasting, the main problem with time series analysis and modeling is that there is only one process implementation at a time (one statistical sample, one time series sample already implemented) that needs to be used to make a forecast for the future. Regardless of the tools used in the analysis method: statistical models, neural networks or fuzzy logic models, the nonstationary time series is divided into certain sections, where it is quasi-stationary with its selective distribution function, and there are parts of the series in which transients occur. The duration of the transition process is determined by both the physical changes and the sample size used for statistical analysis. The parameters of the distribution function are determined based on the analysis of data on the time interval of quasi-stationarity. In particular, nonparametric methods help to restore the probability density based on the observed values. In practice, there are two problems: to determine the time interval of quasi-stationarity and to determine the beginning of the transition period with minimal delay.

To predict the upcoming news, it is advisable to consider time dependences in the flow of events and introduce a piecewise constant approximation of their intensity, using the Bayesian approach and Poisson's distribution to describe future events.

General principles of forecasting news events

As a generalization, we can offer such an approach for forecasting news events based on the analysis of the dynamics of events that have already taken place. The first step is to collect a set of text data from news sites. It is recommended to use the news in English first, because there are more convenient models and tools for further processing. Today, in connection with the reduction of expenditures on information publications, the quality of the media is a particularly important topic that directly and indirectly affects politics, economics and culture. Therefore, proven quality measurements are important for assessing the state of media systems. Unfortunately, this process is not easy due to the double hermeneutics of the social sciences. The ways in which sociologists assess the quality of news media constantly interact with the values of the structures that divide society. This makes the quality of news media a dynamic, conditional and contradictory construction. Nevertheless, by identifying the four main components of media quality (object, ideal, class, and criteria), a fairly clear and adequate media evaluation system can be obtained. The main criteria for the quality of the media are the relevance of news, providing the context in which

an event is, compliance with professional standards and a variety of materials (figure 1).

Figure 2. Media quality criteria

With the help of computational linguistics methods, it is necessary to carry out the following preliminary data processing: delete stop words, perform stemming or lematization and vectorization based on existing dictionaries, create a TF-IDF matrix. Then it is necessary to cluster by thematic groups with the date and time of the news [7]. The input data will be considered a vector representation of the textual description of predicted events, which allows you to find the cosine of the angle between the text and the centroids of thematic clusters derived from the news collection. The change in the value of this cosine in time is considered as a wandering point on the segment [0,1], which contains a trap in the threshold point of the event, where the wandering point can get over time. The minimum value of the allowed cosine metric similarity metric should be considered as a trap. It is necessary to pay attention to the probabilistic schemes of transitions between states in the information space. The parameters of the model can be determined on the basis of the analysis of changes in the structure of the clusters of news over time. The location of the cluster centroid vector and the number of thematic messages per day can be considered as a non-stationary time series. The appearance in the news feed of descriptions of events related to a particular topic, over time, can be considered as the formation of a discrete time series (parameter - the frequency of mentions of the event during the day).

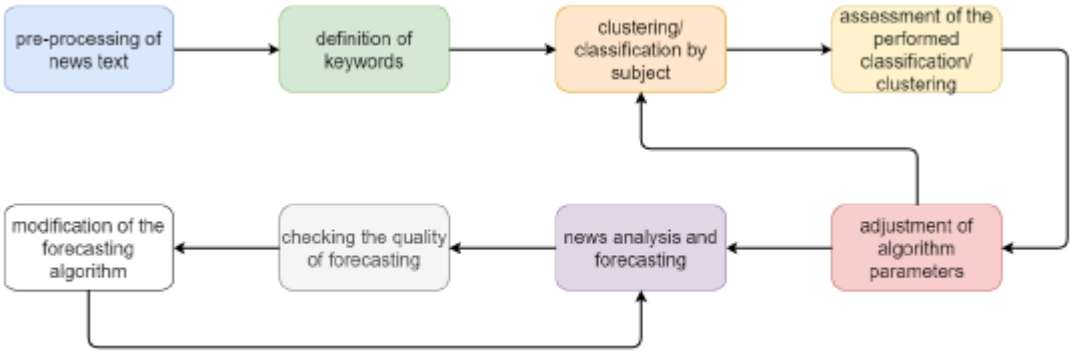

Figure 2. Approximate scheme of work on forecasting events based on

news

Analysis of the dynamic characteristics of a series can be used to predict its change, as well as to calculate the probability of an event occurring over a period of time. Keep in mind that in order to form a time series, you need to solve the problem of selecting from the news feed text messages related to this topic with the highest accuracy. This will ensure that much of the information is not lost in the formation of the time series, for example, the frequency of the event, which will achieve a more accurate definition of the parameters of the time series and will have no effect on its development. As a result, the predicted event can be formed from a set of events from clusters.

Conclusions

The problem of event forecasting based on news processing for the previous period is formulated and defined. It is noted that the problem is complex and may involve the use of such mathematical apparatus as statistical analysis, time series, Markov models and processes, associative rules. An overview of publications on the topic, which methods of solution are used more often than others. The disadvantages of these methods and ways to avoid them are identified. A generalized scheme of problem solving is proposed, which provides for pre-processing of text data, classification or clustering of data for their analysis, application of the prediction method (based on Markov processes, time series, associative rules or neural networks).

REFERENCES

1. Discovering and learning sensational episodes of news events / X. Ao et al. Information Systems. 2018. Vol. 78. P. 68-80. DOI: 10.1016/j.is.2018.05.003.

2. Preethi P. G., Uma V., Kumar A. Temporal Sentiment Analysis and Causal Rules Extraction from Tweets for Event Prediction. Procedia Computer Science. 2015. Vol. 48. P. 84-89. DOI: 10.1016/j.procs.2015.04.154.

3. Anastasiu D. C., Tagarelli A., Karypis G. Document Clustering: The Next Frontier. Data Clustering. 2018. P. 305-338. DOI: 10.1201/9781315373515-13.

4. Realization of a news dissemination agent based on weighted association rules and text mining techniques / C. Huang et al. Expert Systems with Applications. 2010. Vol. 37, № 9. P.

6409-6413. DOI: 10.1016/j.eswa.2010.02.078.

5. Modeling Precursors for Event Forecasting via Nested Multi-Instance Learning / Y. Ning et al. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016. DOI: 10.1145/2939672.2939802.

6. Singh S., Khatri R. Data Mining based Technique for Natural Event Prediction and Disaster Management. International Journal of Computer Applications. 2016. Vol. 139, № 14. P. 34-39. DOI: 10.5120/ijca2016909102.

7. Zhukov D., Andrianova E., Trifonova O. Stochastic Diffusion Model for Analysis of Dynamics and Forecasting Events in News Feeds. Symmetry. 2021. Vol. 13, № 2. P. 257. DOI:

10.3390/sym13020257.

КПІ ім. Ігоря Сікорського

Матеріали Першої Всеукраїнської науково-практичної конференції молодих вчених та студентів «Інженерія програмного забезпечення і передові інформаційні технології» (SoftTech-2021)

2021

22-26 листопада
Україна, Київ

Оргкомітет конференції

Оргкомітет конференції

Голова організаційного комітету: Е.В. Жаріков – в.о. зав. кафедри ІПІ, д.т.н., професор.

Члени організаційного комітету:

П.І.П.	**Посада**
Е.В.Жаріков	В.о. зав. кафедри ІПІ
О.А. Павлов	Професор кафедри ІПІ
І.В. Стеценко	Професор кафедри ІПІ
І.П. Муха	Доцент кафедри ІПІ
Ю.О. Олійник	Доцент кафедри ІПІ
К.І. Ліщук	Доцент кафедри ІПІ
І.В.Баклан	Доцент кафедри ІПІ

Члени програмного комітету:

П.І.П.	**Посада**
В.В.Гнатушенко	Професор НМетАУ
С.А.Бабічєв	Професор ХДУ
В.І.Литвиненко	Професор ХНТУ
Г.В.Рудакова	Професор ХНТУ
О.В. Гавриленко	Доцент кафедри ІСТ
О.Д.Фіногенов	Доцент кафедри ІПІ
О.І.Лісовиченко	Доцент кафедри ІПІ
Т.А.Ліхоузова	Доцент кафедри ІПІ

Перша Всеукраїнська науково-практична конференція молодих вчених та студентів «Інженерія програмного забезпечення і передові інформаційні технології» (SoftTech-2021). Секція кафедри інформатики та програмної інженерії. Матеріали конференції. – Київ. – 2021. 22–26 листопада 2021р. – 198 с.

У збірник включені тези доповідей, які були представлені на конференції «Інженерія програмного забезпечення і передові інформаційні технології» (SoftTech-2021) в секції інформатики та програмної інженерії. В доповідях розглянуті наукові та методичні питання щодо сучасних аспектів інформатики та обчислювальної техніки.

Редакційна колегія:

Баклан І.В. доцент, к.т.н, доцент кафедри ІПІ НТУУ «КПІ ім. Ігоря Сікорського», Муравйова І. М., інженер І категорії кафедри ІПІ НТУУ «КПІ ім. Ігоря Сікорського»

Дизайн титульної сторінки: провідний інженер Майєр З.О. кафедри ІПІ НТУУ «КПІ ім. Ігоря Сікорського»

УДК 681.513.7; 519.688

ПРОЦЮК Ю. В.,
ГАВРИЛЕНКО О. В.

ОГЛЯД ЗАДАЧІ ПРОГНОЗУВАННЯ ПОДІЙ НА ОСНОВІ НОВИН ТА МЕТОДІВ ЇЇ РОЗВ'ЯЗАННЯ

Проведено огляд і порівняння сфер застосування сучасних методів та підходів до прогнозування подій на основі аналізу історичних даних. Обгрунтовано актуальність теми та вказано можливі сфери подальшого застосування результатів прогнозування. Визначено основні етапи роботи з даними для прогнозування подій, а саме попередня обробка, аналіз та побудова прогнозу. Вказано, що асоціативні правила чи марковські процеси можуть допомогти знайти причинно-наслідкові зв'язки між подіями, а часові ряди визначити з заданою ймовірністю період, коли настане подія.

ПРОГНОЗУВАННЯ ПОДІЙ, КОМП'ЮТЕРНА ЛІНГВІСТИКА, ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ, ЧАСОВІ РЯДИ, АСОЦІАТИВНІ ПРАВИЛА, НОВИНИ

A review and comparison of areas of application of modern methods and approaches to event forecasting based on the analysis of historical data. The relevance of the topic is substantiated and possible areas of further application of forecasting results are indicated. The main stages of working with data for forecasting events are identified, namely the preliminary processing, analysis and construction of the forecast. It is stated that associative rules or Markov processes can help to find cause-and-effect relationships between events, and time series to determine with a given probability the period when the event will occur.

EVENT FORECASTING, COMPUTER LINGUISTICS, DATA MINING, TIME SERIES, ASSOCIATIVE RULES, NEWS

1. Вступ

Засоби масової інформації вже давно перестали бути просто джерелом новин щодо подій у світі. Важко заперечувати роль ЗМІ у впливі на людські уми й на взагалі хід подій історій. Все частіше можна помітити, як новини передбачають ті чи інші події, ніби накликають їх. Тому постає питання побудови моделі, що допомагатиме прогнозувати подій в майбутньому базуючись на інформації про минулу.

Виявлення знань з текстів природними мовами є одним з найважливіших питань інтелектуального аналізу даних. Це може розв'язати глобальну проблему пошуку прихованих закономірностей, що дозволяють прогнозувати можливі майбутні новинні події та створювати моделі впливу на різноманітні економічні та соціальні процеси, як-от виникнення економічної кризи чи негативних соціальних явищ. Розвиток нових математичних моделей для прогнозування новинних подій на основі аналізу текстів природної мови є особливо актуальним питанням розвитку інтелектуального аналізу даних та виявлення знань, бо дозволить не лише виявляти приховані закономірності й знання, а й передбачати на їх основі наступні подій в майбутньому. Аналіз текстів та вилучення з них даних для використання в моделях прогнозування може базуватися на відомих методах комп'ютерної лінгвістики та інтелектуального аналізу тексту.

2. Аналіз останніх публікацій

Загалом, аналіз новин для прогнозування майбутніх подій – тема малодосліджена через свою складність. Серед іншого в роботі [1] за допомогою методів обробки природної мови досліджувались особливості виникнення взаємопов'язаних сенсаційних новин на основі аналізу тексту. У статті досліджувалися закономірності появи пар подій, що настали, у просторі новин і як спрогнозувати, що інша подія настане після першої. Методи комп'ютерної лінгвістики використовувалися, щоб знайти причинно-наслідкові зв'язки між подіями з їх тестових описів.

Дослідження [2] присвячене темі

виявлення причинно-наслідкових зв'язків між подіями в соцмережах, щоб спрогнозувати тональність події та час між настанням різних подій. Спершу вибираються повідомлення за проміжок часу, з них вибираються ключові слова, що використовуються для визначення тональності повідомлення – позитивної, негативної чи нейтральної. Для визначення тональності слів використовується класифікатор, що навчається на основі методу опорних векторів. Надалі будуються причинно-наслідкові зв'язки між ключовими словами, для цього використовується метод асоціативних правил, що визначає правила «якщо-то» з даних. Останнім етапом є прогнозування подій з використанням часового аналізу повідомлень та розрахунку причинно-наслідкових зв'язків.

Робота [3] бере за основу лінгвістичний аналіз і статистичне моделювання твітів для автоматичного визначення тем, що обговорюються у великих містах. Щоб виокремити теми, рекомендується застосовувати тематичне моделювання, для цього своєю чергою текст твітів розбивався на токени з допомогою спеціального токенізатора та часткового мовного тегера. При цьому смайлики розглядалися як окремі токени, що несуть певне змістовне навантаження. Для моделювання теми також аналізується семантичний контент, що описує емоційний стан автора твіта.

Проблема впливу заголовків новин на поведінку інвесторів та зміни на фінансовому ринку висвітлена у роботі [4]. Модель, що заснована на зважених асоціативних правилах, визначає чи достатньо важливою є новина для інвесторів. Під час навчання на майбутніх подій. За даними колекції потокових новин з відкритих джерел був розроблений вкладений підхід для прогнозування значних публічних подій та протестів. Сильні сторони вказаного підходу доводяться емпіричною оцінкою, що полягає в фільтрації потенційних передвісників для точного прогнозування характеристик подій громадянських заворушень і в самому прогнозування настання подій з перевагою в часі виконання.

Автори дослідження [6] представляють модель прогнозування нещасних випадків, що закінчилися смертю, а також стихійних лих. Автори зібрали текстові повідомлення з системи Google про катастрофи. Отримані текстові документи оброблялися за допомогою методів комп'ютерної лінгвістики й хибні результати відсіювалися за допомогою навченого баєсівського класифікатора. Після збору даних була проведена семантична кластеризація цих даних. Матриця переходів була побудована з ключових слів, які використовувалися при зборі даних. Матриця спостереження ж була збудована зі згрупованих подій. Обидві матриці подавалися на вхід прихованої марковської моделі для прогнозування. Щоб спрогнозувати нову подію із вказаною темою, потрібно створити модель її формування на основі опису її часового ряду, а потім знайти функцію щільності розподілу її параметрів. При складанні прогнозів основна проблема аналізу та моделювання часового ряду полягає в тому, що в будь-який момент часу існує лише одна реалізація процесу (одна статистична вибірка, одна вибірка часового ряду, який вже реалізовано), які потрібно використати для створення прогнозу на

реальних даних алгоритм зважених асоціативних правил знаходить терміни, що часто й одночасно з'являються в заголовках новин. Термін з'являється в заголовках новин за день кілька раз в певний день упродовж певної кількості днів. І вага впливу терміну визначається через те, як сильно змінилася ціна акції за період з врахуванням частоти вживання даного терміну. Ці ваги дозволяють визначити чи впливають ті чи інші терміни на результати торгівлі.

Робота [5] пропонує методи, що розв'язують проблему ідентифікації подій, що є передвісниками та власне передбачень майбутнє. Незалежно від того, які інструменти використовуються в методі аналізу: статистичні моделі, нейронні мережі чи моделі нечіткої логіки, нестаціонарний часовий ряд розбивається на окремі області, де він є квазістаціонарним зі своєю вибірковою функцією розподілу, і є частини ряду, в яких відбуваються перехідні процеси. Тривалість перехідного процесу визначається як фізичними змінами, так і розміром вибірки, що використовується для статистичного аналізу. Параметри функції розподілу з'ясовуються на основі аналізу даних на часовому інтервалі

квазістаціонарності. Зокрема непараметричні методи допомагають відновити щільність імовірності на основі спостережуваних значень. Практично виникає дві проблеми: визначити часовий інтервал квазістаціонарності та визначити початок перехідного періоду з мінімальною затримкою.

Для прогнозування новин, що настануть, доцільно розглянути часові залежності в потоках подій і ввести кусково-постійну апроксимацію їх інтенсивності, використавши для цього баєсівський підхід та розподіл Пуассона для опису майбутніх подій.

3. Узагальнення підходу прогнозування новинних подій

Загалом можна запропонувати такий підхід для прогнозування новинних подій на основі аналізу динаміки подій, що вже відбулися. Найперше потрібно зібрати набір текстових даних з новинних сайтів. Рекомендується спочатку використовувати новини англійською мовою, оскільки саме для неї існує більша кількість зручних моделей та інструментів подальшої обробки. За допомогою методів комп'ютерної лінгвістики варто провести таку обробку: виконати лематизацію та векторизацію на основі наявних словників, створити матрицю TF-IDF, кластеризувати за тематичними групами з датуванням за часом новини [7]. Вхідними даними вважатимемо векторне представлення текстового опису прогнозованих подій, що дозволяє знайти значення косинуса кута між ними та центроїдами тематичних кластерів, що отримані з колекції новин. Зміна значення даного косинуса в часі розглядається як блукання точки на відрізку [0,1], що містить пастку в пороговій точці реалізації події, куди точка блукання може потрапити з часом. Як пастку варто розглядати мінімальне значення дозволеної косинусної метрики подібності векторів. Необхідно звернути увагу на ймовірнісні схеми переходів між різними станами в інформаційному просторі. Параметри моделі можуть бути визначені на основі аналізу змін у структурі наявних тематичних кластерів новин в часі. Положення вектора центроїда кластера та кількість повідомлень на тему протягом доби можна розглядати як нестаціонарний часовий ряд. Появу з часом в стрічці новин описів подій певного типу, що пов'язані з певною темою, з часом можна розглядати як формування дискретного часового ряду (параметр якого – частота згадок події упродовж дня).

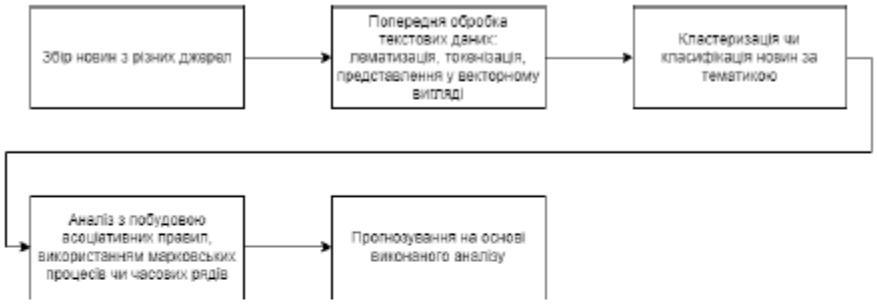

Рис. 1 – Орієнтовна схема роботи над прогнозуванням подій на основі новин

Аналіз динамічних характеристик даного ряду може бути використаний для прогнозування його розвитку, а також для розрахунку ймовірності подій, що відбуваються протягом заданого інтервалу часу. Важливо пам'ятати, що для формування часового ряду потрібно розв'язати задачу вибору з новинної стрічки текстових повідомлень, що належать до даної теми з максимально високою точністю. Це гарантуватиме, що значна частина інформації не буде втрачена при формуванні часового ряду, наприклад, по частоті настання події, що дозволить досягнути більш точного визначення параметрів часового ряду і не вплине на прогноз його розвитку. В результаті прогнозована подія може бути сформована з набору подій з кластерів.

Висновки

Сформульовано та визначено задачу прогнозування подій, що базується на обробці новин за попередній проміжок часу. Зазначено, що задача є комплексною та може передбачати використання такого математичного апарату, як статистичний аналіз, часові ряди, марковські моделі та процеси, асоціативні правила. Проведено огляд публікацій з теми, вказано, які методи розв'язання використовуються частіше інших. Визначено недоліки названих методів та способи, як їх можна уникнути. Запропоновано узагальнену схему розв'язку задачі, що передбачає попередню обробку текстових даних, класифікацію чи кластеризацію даних для їх аналізу, застосування методу прогнозування (заснованого на марковських процесах, часових рядах, асоціативних правилах чи нейронних мережах).

Перелік посилань

1. Discovering and learning sensational episodes of news events / [X. Ao, P. Luo, C. Li та ін.]. // Information Systems. – 2018. – №78. – С. 68–80.
2. Preethi P. Temporal Sentiment Analysis and Causal Rules Extraction from Tweets for Event Prediction / P. Preethi, V. Uma, A. Kumar. // Procedia Computer Science. – 2015. – №48. – С. 84–89.
3. Gerber M. Predicting crime using Twitter and kernel density estimation / Matthew S. Gerber. // Decision Support Systems. – 2014. – №61. – С. 115–125.
4. Realization of a news dissemination agent based on weighted association rules and text mining techniques / [C. Huang, J. Liao, D. Yang та ін.]. // Expert Systems with Applications. – 2010. – С. 6409–6413.
5. Modeling Precursors for Event Forecasting via Nested Multi-Instance Learning / Y.Ning, S. Muthiah, H. Rangwala, N. Ramakrishnan. // the 22nd ACM SIGKDD International Conference. – 2016. – С. 1095–1104.
6. Singh S. Data Mining based Technique for Natural Event Prediction and Disaster Management / S. Singh, R. Khatri. // International Journal of Computer Applications. – 2016. – №139. – С. 34–39.
7. Zhukov D. Stochastic Diffusion Model for Analysis of Dynamics and Forecasting Events in News Feeds / D. Zhukov, E. Andrianova, O. Trifonova. // Symmetry. – 2021. – №13.

Додаток Б Додаткові графічні матеріали

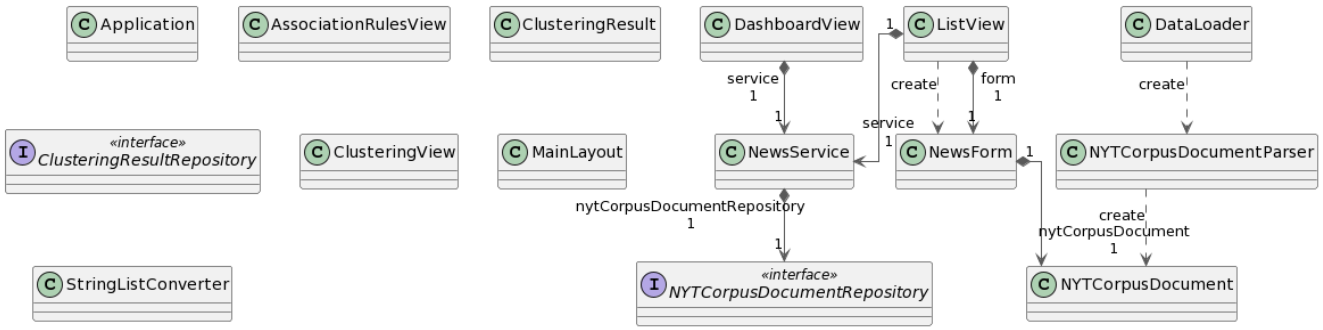

Схема структурна класів інформаційної системи

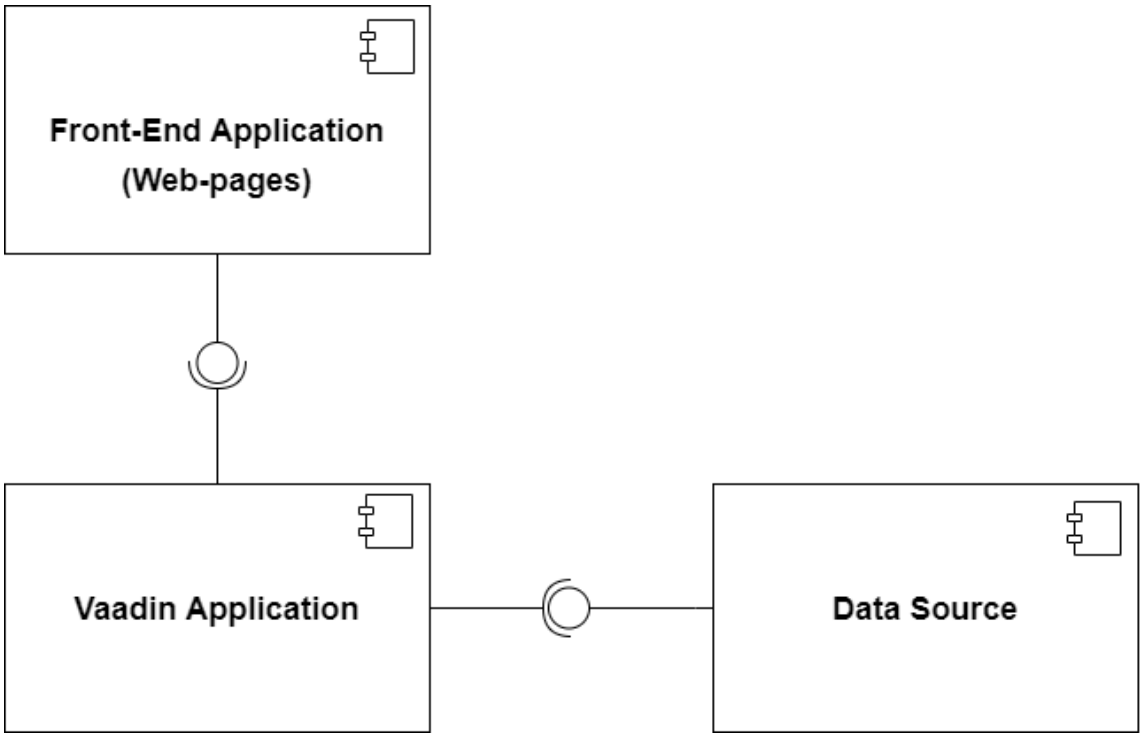

Схема структурна компонентів інформаційної системи

Ключові слова по кластерах (метод к-середніх з використанням FastText). Частина 1

Ключові слова по кластерах (метод к-середніх з використанням FastText). Частина 2

Ключові слова по кластерах (агломеративна кластеризація). Частина 1

Ключові слова по кластерах (агломеративна кластеризація). Частина 2

Додаток В Набір самостійно визначених стоп-слів

About, after, against, all, also, among, and, another, any, are, around, back, because, become, been, before, began, being, between, both, but, called, came, can, city, come, could, did, does, down, during, each, early, end, even, every, far, few, first, five, for, found, four, from, get, go, going, good, group, had, has, have, he, her, here, him, his, how, into, its, just, know, last, left, less, life, like, little, long, made, make, man, many, may, might, more, most, much, never, new, next, not, now, off, often, old, one, only, other, our, out, over, own, part, play, put, recent, said, same, say, second, see, set, several, she, should, show, since, some, still, such, take, than, that, the, their, them, then, there, these, they, think, this, those, three, through, time, told, too, took, two, under, until, used, very, want, was, way, well, were, what, when, where, which, while, who, whose, will, with, without, won, work, works, would, year, years, you, young, according, ago, almost, along, already, although, always, asked, aug, away, based, became, best, better, big, case, different, enough, full, general, great, high, important, include, including, known, large, late, later, lead, lot, making, men, must, near, need, news, others, past, present, real, recently, right, say, small, something, talk, thing, third, though, top, toward, woman, women, yet.

Додаток Г Ліцензія на використання вхідного набору даних

User License Agreement for The New York Times Annotated Corpus (LDC2008T19)

Application by an Organization to use The New York Times Annotated Corpus (LDC2008T19) distributed by the Linguistic Data Consortium (LDC)

NTUU KPI ("User"), an organization engaging in language education and research agrees to use the text data designated as **The New York Times Annotated Corpus (LDC2008T19)** (the "Data") and distributed by the LDC subject to the following understandings, terms and conditions.

1. Permitted and Prohibited Uses

1.1. The Data may only be used for non-commercial linguistic education, research and technology development, including but not limited to information retrieval, document understanding, machine translation or speech recognition.

1.2. User shall not publish, retransmit, display, redistribute, reproduce or commercially exploit the Data in any form, except that User may include limited excerpts from the Data in articles, reports and other documents describing the results of User's linguistic education, research and technology development.

2. Copyright Notice and Disclaimer

2.1. The Data is owned or controlled by The New York Times Company and is protected by applicable copyright law. In no event shall User publish, retransmit, display, redistribute, or otherwise reproduce any or all of the Data in any format to anyone, except as allowed in Section 1 of this agreement, in which case User shall provide The New York Times Company with copyright attribution notice as follows:

"© [appropriate year] The New York Times Company, used with permission".

2.2. USER ACKNOWLEDGES AND AGREES THAT **THE NEW YORK TIMES ANNOTATED CORPUS (LDC2008T19)** IS PROVIDED ON AN "AS-IS" BASIS AND THAT LDC AND ITS HOST INSTITUTION THE UNIVERSITY OF PENNSYLVANIA MAKE NO REPRESENTATIONS OR WARRANTIES OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR CONFORMITY WITH WHATEVER DOCUMENTATION IS PROVIDED. IN NO EVENT SHALL LDC OR ITS HOST INSTITUTION BE LIABLE FOR SPECIAL, DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, INCIDENTAL OR OTHER DAMAGES, LOSSES, COSTS, CHARGES, CLAIMS, DEMANDS, FEES OR EXPENSES OF ANY NATURE OR KIND ARISING IN ANY WAY FROM THE FURNISHING OF, OR USER'S USE OF, **THE NEW YORK TIMES ANNOTATED CORPUS (LDC2008T19)**.

User shall send a signed copy of this agreement by facsimile to LDC, fax number (+1) 215 573-2175. Alternately, User shall email an electronic version of the signed agreement to LDC at ldc@ldc.upenn.edu.

For NTUU KPI
Signature Yurii Protsiuk 
Date 06.04.2022
Name Yurii Protsiuk
Title