УДК 004.9

ТЕХНОЛОГІЯ АНАЛІЗУ УКРАЇНОМОВНИХ ТВІТІВ ДЛЯ ПРОГНОЗУВАННЯ ЗМІНИ ДИНАМІКИ ГРОМАДСЬКОЇ ДУМКИ НА ОСНОВІ МАШИННОГО НАВЧАННЯ

Прокіпчук О. А. – магістр кафедри «Інформаційні системи та мережі», Національний університет «Львівська політехніка», Львів, Україна.

Висоцька В. А. – канд. техн. наук, доцент, доцент кафедри «Інформаційні системи та мережі», Національний університет «Львівська політехніка», Львів, Україна.

АНОТАЦІЯ

Актуальність. Автоматизація дослідження громадської думки дозволить не тільки зменшити кількість ручної праці, а й отримувати часові зрізи результатів без додаткових зусиль. Оскільки потрібно уникнути прямої взаємодії з респондентами, громадську думку необхідно аналізувати на основі джерел її вільного вираження. Соціальні мережі чудово підходять на цю роль, так як там люди вільно публікують свої думки або емоційно правдиво реагують на опубліковану інформації щодо певних подій. Статистика показує, що даних із соціальних мереж недостатньо для отримання повноцінного результату, бо чималий відсоток людей не користуються соціальними мережами. Проте автоматизація дослідження навіть такого прошарку населення уже є хорошим результатом для аналізу динаміки змін громадської думки відповідно подій в країні/світі та відповідно для корегування в подальшому процесів державного управління.

Мета дослідження – розроблення технології аналізу україномовного потоку контенту в соціальних мережах для дослідження громадської думки на основі знаходження кластеризованих тематичних груп твітів.

Метод. В статті розроблено технолгію пошуку трендів твітів на основі кластеризації, що формує потік даних у вигляді коротких репрезентацій кластерів та їхньої популярності для подальшого дослідження громадської думки. Описано ефективний підхід збору твітів, їх фільтрації, очищення та попереднього опрацювання на основі порівняльного аналізу алгоритмів Bag of Words, TF-IDF та BERT. Визначено вплив стемінгу та лематизації на якість отриманих кластерів. А також знайдено оптимальні поєднання методів кластеризації (цій K-Means, Agglomerative Hierarchical Clustering та HDBSCAN) та векторизації твітів на основі аналізу 27 кластеризацій однієї вибірки даних. Обрано спосіб подання кластерів твітів у короткому форматі.

Результати. Найкращі результати показали алгоритми, що використовують Відстань Левенштейна, тобто fuzz sort, fuzz set та levenshtein. Дані алгоритми швидко здійснюють перевірки, мають більшу різницю подібностей, тож можна точніше визначити межу подібності. Згідно з результатами проведених кластеризацій, оптимальними рішеннями є використання алгоритму кластеризації HDBSCAN та алгоритму векторизації BERT для досягнення найточніших результатів, та використання K-Means разом із TF-IDF для досягнення найкращої швидкодії із оптимальним результатом. Для зменшення часу виконання можна застосовувати стемінг.

Висновки. В даному дослідженні експериментально знайдено оптимальні варіанти для порівняння відбитків кластерів серед таких методів пошуку подібності: Fuzz Sort, Fuzz Set, Levenshtein, Jaro Winkler, Jaccard, Sorensen, Cosine, Sift4. У деяких алгоритмів середня подібність відбитків сягає вище 70%. Знайдено 3 ефективні інструменти для порівняння їхньої подібності, так як вони показують достатню відмінність між порівняннями подібних та різних кластерів (> 20%). На основі обраних ефективних методів, успішно проведено аналіз трендів для 90 000 твітів за 7 днів для 5 тем тижня за допомогою K-Means та TF-IDF для кластеризації та векторизації, а також fuzz sort для порівняння відбитків кластерів із межою подібності 55%.

КЛЮЧОВІ СЛОВА: твіт, українська мова, громадська думка, тренд, кластеризація, стеммінг, лематизація, подібність кластерів.

АБРЕВІАТУРА

BERT – Bidirectional Encoder Representations from Transformers;

BOW – Bag of Words;

HDBSCAN – Hierarchical Density-Based Spatial Clustering of Applications with Noise;

NLP – Natural Language Processing.

НОМЕНКЛАТУРА

S – система аналізу твітів;

I – множина вхідних даних;

O – множина вихідних даних;

R – основні правила опрацювання вхідних даних;

U – параметри опрацювання вхідних даних;

N – нейронна мережа;

α – оператор скачування вхідних даних;

β – оператор опрацювання вхідних даних;

γ – оператор кластеризації твітів;

μ – оператор ідентифікації тематичних твітів;

χ – оператор формування датасетй твітів;

ω – оператор зняття відбитків кластерів твітів;

λ – оператор злиття кластерів твітів;

i_1 – множина данних із соціальної мережі;

i_2 – сховище даних твітів;

i_3 – словники укаїномоних слів;

i_4 – множина тематичних ключових слів твітів;

o_1 – періодичні запити на збір твітів;

o_2 – реузльтат кластеризації;

o_3 – результат злиття кластерів;

r_1 – правила збору даних з соціальних мереж;

r_2 – правила NLP україномовних твітів;

r_3 – правила кластеризації твтів;

DOI 10.15588/1607-3274-2023-2-11

r_4 – правила злиття кластерів твітів;
u_1 – множина умов збору твітів в соціальні мережі;
u_2 – множина вимог фільтрування твітів від шуму;
u_3 – множина умов опрацювання твітів;
u_4 – множина умов кластеризації твітів;
u_5 – множина вимог формування висновків.

ВСТУП

Дослідження громадської думки є невід'ємним елементом зворотного зв'язку народу до держави. Станом на сьогодні, основними методами збору громадської думки в Україні є опитування громадян в офлайн чи онлайн форматах. Проведенням цих опитувань зазвичай займаються соціологічні центри. В більшості випадків такий процес є частково автоматичний (проведення анкетування з подальшим машинним опрацюванням). Недоліком такого опитування є неточність отриманих даних за рахунок не охоплення великої кількості прошарків населення та різних соціальних груп, хаотичність та довготривалість збору даних, що приводить ще і до втрати актуальності отриманих результатів. Крім того часто в таких анкетах люди дають не зовсім правдиву інформацію. Аналіз же реакцій соціальних груп на ту чи іншу подію/інформацію в соціальних мережах не скриває їх справжньої емоційної реакції. Автоматичний збір та аналіз подібної інформації приводить до отримання оперативних актуальних даних по певних проблемних питаннях для подальшого дослідження відповідної громадської думки. В роботі розглядатиметься технологія аналізу інформаційного потоку в соціальних мережах, що містять тренди україномовних твітів. Їх разом із мета-даними доцільно використовувати для автоматизації дослідження громадської думки. Наприклад, в умовах війни, важливо знати настрої українців для формування реакцій державного управління та кроків/алгоритмів розвитку/відбудови країни. Окрім того, автоматизація процесу дозволить створити незалежне загальнодоступне джерело інформації із хронологічними даними, наприклад, для можливості прогнозування реакцій населення на внесення змін в процес державного управління.

Метою дослідження є розроблення технології аналізу україномовного потоку контенту в соціальних мережах для дослідження громадської думки на основі знаходження кластеризованих тематичних груп твітів. Для досягнення мети були поставлені такі завдання:

– визначити ефективний підхід формування вибірки та попереднього опрацювання україномовних твітів на основі NLP-методів;

– провести експериментальне дослідження методів та інструментів для створення кластерів україномовних твітів для визначення оптимального їх поєднання при отриманні найкращих результатів на великих обсягах вхідних даних;

– розробити метод розрахунку популярності кластерів, відображення кластерів у короткому форматів та об'єднання кластерів, створених у різні моменти часу на основі проведення експериментальних апробацій;

– здійснити аналіз результатів експериментальної апробації запропонованої технології аналізу твітів.

Об'єкт дослідження – процеси ідентифікації та аналізу україномовних твітів для прогнозування зміни динаміки громадської думки. Предмет дослідження – методи та засоби ідентифікації та аналізу україномовних твітів для прогнозування зміни динаміки громадської думки із застосуванням оптимального конвеєру (pipeline) на основі NLP-методів, кластерного аналізу, вибору та генерування ознак, алгоритмів машинного навчання, в умовах наявності великих за обсягом корпусів анотованих даних.

1 ПОСТАНОВКА ПРОБЛЕМИ

Систему аналізу україномовних твітів S подано імітаційною моделлю через кортежем:

$$S = < I, O, R, U, N, \alpha, \beta, \gamma >,$$

Де $I = \{i_1, i_2, i_3, i_4\}$, $O = \{o_1, o_2, o_3\}$, $R = \{r_1, r_2, r_3, r_4\}$, $U = \{u_1, u_2, u_3, u_2, u_3\}$.

Основними процесами моделі аналізу україномвоних твітів в соціальній мережі є «Збір твітів», «NLP твітів», «Машине навчання» та «Формування висновків».

Процес «Збір твітів» опишемо суперпозицією:

$$C_{AU} = \mu \circ \beta \circ \alpha,$$
$$C_{AU} = \mu(\beta(\alpha(i_1, i_2, i_4), r_1, u_1), u_2).$$

Процес «NLP твітів»опишемо суперпозицією:

$$C_{CU} = \chi \circ \beta \circ \alpha, \text{ тобто}$$
$$C_{CU} = \chi(\beta(\alpha(C_{AU}, i_2, i_3, i_4), r_1, u_3), r_2).$$

Процес «Машине навчання» опишемо як:

$$C_{UL} = \omega \circ \gamma \circ \beta \circ \alpha,$$
$$C_{UL} = \omega(\gamma(\beta(\alpha(C_{CU}, i_2), i_3), u_4), r_3).$$

Процес «Формування висновків» опишемо як:

$$C_{US} = \lambda \circ \gamma \circ \beta \circ \alpha,$$
$$C_{US} = \lambda(\gamma(\beta(\alpha(C_{US}, i_2), i_4), u_5), r_4).$$

Аналіз здійснюється за допомогою класифікації чи кластеризації, що групують повідомлення за певними критеріями. Хоч збір інформації й відбувається автоматично, проте все ще необхідна реалізація таких досліджень згідно конкретних тем, та відповідна обробка результатів. Також суттєво впливає на результати дослідження ефективність опрацювання відповідної регіональної мови. Ще одним з найважливіших критеріїв подібнрої технології є можливість збору часових даних та їх періодичність.

DOI 10.15588/1607-3274-2023-2-11

2 АНАЛІЗ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

Переважно, соціальне опитування є періодичним процесом, що вимагає залученості як респондентів, так й працівників соціологічних центрів для організації опитування та аналізу результатів. Хоч онлайн системи дозволяють спростити цей процес, він все ще потребує чималої ручної праці, а для отримання часового зрізу громадської думки необхідно здійснити цілу низку повторних опитувань [1]. Автоматизація дослідження громадської думки дозволить не тільки зменшити кількість ручної праці, а й отримувати часові зрізи результатів без додаткових зусиль. Оскільки потрібно уникнути прямої взаємодії з респондентами, громадську думку необхідно аналізувати на основі джерел її вільного вираження. Соціальні мережі чудово підходять на цю роль, так як там люди вільно публікують свої думки або емоційно правдиво реагують на опубліковану інформації щодо певних подій. Статистика показує, що даних із соціальних мереж недостатньо для отримання повноцінного результату, бо чималий відсоток людей не користуються соціальними мережами [2]. Проте автоматизація дослідження навіть такого прошарку населення уже є хорошим результатом для аналізу динаміки змін громадської думки відповідно подій в країні/світі та відповідно для корегування в подальшому процесів державного управління. Існують приклади успішного дослідження громадської думки в соціальних мережах. Наприклад, таким дослідженням громадської думки є аналіз інформації щодо COVID-19 в Китаї на основі соціальної мережі Weibo [3] або вакцинації в Італії на основі мережі Twitter [4]. Такі дослідження проводять для конкретних тем та визначають ставлення суспільства до них. В дослідженні щодо халяль-туризму [5] чудово реалізований збір часових даних твітів. Проте дослідження все ще здійснено щодо певної теми, а кількість проаналізованих твітів не перевищує 100 тисяч. Автоматизована система, що дозволить аналізувати довільні теми із врахуванням часу для цілої країни повинна бути достатньо оптимізована для того, щоб опрацьовувати оперативно, своєчасно та швидко мільйони твітів та оперувати тисячами різних тем. Для забезпечення постійного аналізу необхідно налаштувати збір та дослідження вхідного інформаційного потоку контенту в межах конкретних тем суспільства. І бажано, щоб кількість таких тем була якщо не десятками, а сотнями. Потік тем повинен містити достатньо інформації про себе у вигляді маркерів (ключові слова, часові дані як дати/періоди, основні тези, територіальність/регіональність тощо) для можливості всебічного дослідження громадської думки від популярності теми, її емоційного забарвлення (позитивно, негативно, нейтрально) до можливості прогнозування зміни думки в часі в залежності від подій в межах конкретної теми.

3 МАТЕРІАЛИ ТА МЕТОДИ

Для розроблення технології аналізу україномовних твітів організовують пайплайн (англ. Pipeline, конвеєр) із процесів, що виконуватиметься із певною періодичністю (рис. 1).

Для підтримки постійного доступу до актуальних в часі даних періодичність виконання паплайна повинна становити як мінімум 1 раз на добу. Інтерактивність аналізу дещо змінює звичний процес опрацювання інформаційного потоку контенту за великий період часу та додає декілька необхідних кроків в пайплайн. Загалом алгоритм складається із 5 кроків. Розглянемо кожен із них детальніше.

1. Завантаження твітів. Найоптимальнішим варіантом збору твітів є використання Twitter API [6]. Це потужний інструмент, що дозволяє ефективно завантажувати твіти та використовувати фільтри пошуку. Саме вони дозволять значно скоротити час пошуку та зменшити кількість подальших фільтрацій.

Першим таким фільтром є «-is:retweet», що дозволяє виключити із списку результатів ретвіти. Велика частка ретвітів містить нерелевантну для аналізу інформацію.

Для отримання твітів українською мовою слід використати фільтр «lang:uk». Після цього абсолютна більшість результатів дійсно надходить українською мовою. Twitter API не завжди повертає твіти лише українською. Трапляються твіти на інших мовах, зокрема на російській, із невеликими вкрапленнями української. Включення таких твітів до аналізу погіршить якість результатів. Для покращення вибірки можна додати географічне обмеження на твіти, або виключити з пошуку твіти, що містять певні літери інших алфавітів, наприклад «-ы».

Можна здійснювати пошук твітів, опублікованих у будь-який час доби або лише у денний час для покращення якості вибірки. Далі відбувається фільтрація без участі Twitter API. Слід видалити із вибірки твіти довжиною менше 3–5 слів та твіти, що не містять слів, а лише символи, числа, емодзі чи хеш-теги. Покращити вибірку можна, видаливши дублікати. Оскільки виконувати n^2 операцій може бути дорого, оптимізувати алгоритм можна видаляючи лише ті дублікати, що опубліковані у дуже вузькому проміжку часу.

Рисунок 1 – Пайплайн аналізу твітів

DOI 10.15588/1607-3274-2023-2-11

2. Попереднє опрацювання твітів

2.1. Очистка тексту від шуму. Очистка дозволить позбутись усього зайвого, що може спотворити кінцевий результат. Очистка складається із видалення емоджі, посилань, тегів, хеш-тегів, розділових знаків, чисел, дубльованих пробілів, нетекстових символів та слів із буквами, що не належать українському чи англійському алфавітам. Також потрібно привести текст до нижнього регістру.

2.2. Видалення стоп-слів. Стоп-слова (наприклад, службові) не повинні впливати на результат аналізу тексту. Ці слова не несуть сенсу для аналізу та перевантажують текст зайвою інформацією. При подальшому аналізі тексту, стоп-слова зменшують значущість дійсно важливих для аналізу слів [7]. Українська мова є дуже багатою на такі конструкції. Для приклада бібліотека production рівня SpaCy використовує список, що складається із понад 450 слів [8]. Також зустрічаються й варіанти, що налічують до 2000 слів [9]. Окрім стоп-слів української мови в цілому, до списка слід додати власні назви, що спотворюють результати аналізу такі, як назви новинних організацій. Ці назви містяться в кожному твіті організації та не несуть ніякого значення. Інші ж власні назви можуть суттєво позитивно відгравати на результат пошуку, наприклад Володимир Зеленський, Україна, Херсон, НАТО тощо.

2.3. Стемінг та Лематизація це процеси, що дозволяють проводити злиття подібних слів шляхом приведення різних форм слова до певної основи. Стемінг здійснює приведення до основи слова шляхом відкидання частин, а Лематизація приводить слово до своєї базової форми, спираючись на визначенні частини мови. Користь цих процесів в тому, що вони дозволяють зменшити вимірність (розмірність) вхідного тексту й, тим самим, зробити об'єднання твітів за одною темою більш релевантним. Проте необхідність використання даного опрацювання тексту залежить від точності реалізації відповідних інструментів для української мови. Якщо здійснити злиття для неключових слів, але для ключових слів провести цей процес некоректно, тоді можна збільшити вагу неключових слів над ключовими.

3. Кластеризація твітів є процесом машинного навчання без вчителя, мета якого полягає в розбитті вхідної множини на кластери так, що всередині кластерів знаходяться подібні елементи і водночас мають більшу відмінність із елементами інших кластерів. У визначенні трендів україномовних твітів цей процес необхідний для об'єднання твітів за спільними темами. Такий підхід повинен більше враховувати семантичне наповнення твітів, ніж групування за вмістом ключових слів.

Алгоритми кластеризації працюють із числовими векторами, проте твіти є текстовими даними. Спочатку потрібно перетворити текст у числовий вектор. Для цього існують алгоритми векторизації тексту. В межах даного дослідження розглянемо три алгоритми: BOW, TF-IDF та BERT.

1. BOW є найбільш простим та прямолінійним підходом до векторизації тексту. Метод полягає в підрахуванні слів у тексті, а значення для відповідних слів у векторі позначаються як 1 або 0, що означає присутність або відсутність певного слова у твіті. Такий підхід до подання тексту «як є» може показати як хороший результат, так і поганий, розмивши межі кластерів.

2. TF-IDF дуже популярний алгоритм векторизації, що покращує підхід BOW шляхом введення оберненої частоти документа (inverse document frequency), котра впливатиме на важливість слів, враховуючи їхню частоту в усій вибірці твітів. Така векторизація зможе відділяти широковживані від рідкісних слів, що дозволить якісніше визначити ключові слова при кластеризації.

3. BERT – це сімейство моделей мови, що опублікована у 2018 році [10] та здобула певного визнання. Попередньо треновані моделі BERT можна застосовувати для векторизації тексту з високою якістю. В даному дослідженні використовуватиметься модель, натренована на статтях вікіпедії різних мов, що включають українську [11].

В даному дослідженні проаналізуємо алгоритми кластеризації K-Means або К-Середніх, Agglomerative Hierarchical Clustering та HDBSCAN.

1. K-Means є одним із популярниих алгоритмів, підхід якого базується на підборі центрів кластерів так, щоб сума квадратів відстаней їхніх елементів до центрів була мінімальною. Для роботи алгоритму потрібно визначити кількість кластерів завчасно. Для автоматизації вибору кількості кластерів можна скористатись методами ліктя або силуетної оцінки.

2. Agglomerative Hierarchical Clustering – ієрархічна кластеризація, що імплементує підхід знизу-вгору [12]. Процес починається із визначенням кожного елементу, як центру свого кластеру з їхнім злиттям в подальшому.

3. HDBSCAN – це метод кластеризації, що базується на щільності [13]. Така кластеризація об'єднує в кластери елементи із високою щільністю. Усі елементи, що не є щільними до жодного із кластерів вважаються шумом. HDBSCAN на відміну від DBSCAN дозволяє знаходити кластери із різними щільностями. Цей підхід дозволить кластеризувати твіти інакшим шляхом, ніж К-Середніх, що важливо для пошуку оптимального підходу, проте невдала векторизація може спричинити багато шуму.

4. Зняття відбитків кластерів. В залежності від розміру вибірки, розмір одного кластеру може сягати більше 10 тисяч повідомлень. При щоденному аналізі кількість таких кластерів дуже швидко виросте до такого числа, коли зберігання та обробка кластерів стане занадто дорогим процесом, щоб його підтримувати. Для вирішення цієї проблеми застосований термін «відбиток кластеру». Це скорочена версія, що дозволяє репрезентувати основний зміст кластеру й при цьому бути достатньо

DOI 10.15588/1607-3274-2023-2-11

коротким для зручного зберігання та швидкої обробки. Реалізувати такий відбиток можна різними способами, наприклад зберігати твіти у вигляді скороченого BOW. В межах даного дослідження вирішено створювати відбитки як набори ключових слів кластерів. Такий формат зручно зберігати, застосовувати в подальших обробках і він не втрачає основного змісту кластеру. Існує багато інструментів отримання ключових слів з тексту. В даній роботі використовуватимемо інструмент YAKE [14], оскільки він є простий у використанні.

5. Злиття кластерів. Оскільки процес створення кластерів відбувається не одноразово, а щоденно, потрібно поєднувати результати отриманої кластеризації із кластерами, отриманими у попередні дні. Цей процес є злиттям кластерів і виконується на основі відбитків. Два відбитки потрібно перевірити на подібність. Якщо подібність перетинає певну межу, тоді відбитки кластерів та їхні метадані, що включають кількість твітів кластеру, зливаються в один. Найпростішим варіантом злиття є злиття лише метаданих без модифікації відбитку. Більш просунутий процес передбачає об'єднання відбитків шляхом вибірки ключових слів із кожного з них. Вибірка може бути випадковою або на основі ваги ключових слів. Існує багато інструментів знаходження подібності тексту. В даному дослідженні експериментально знайдено оптимальні варіанти для порівняння відбитків кластерів серед таких методів пошуку подібності: Fuzz Sort, Fuzz Set, Levenshtein, Jaro Winkler, Jaccard, Sorensen, Cosine, Sift4.

4 ЕКСПЕРИМЕНТИ

Першим важливим моментом, що потребує це дослідження, є стемінг та лематизація вхідного тексту українською мовою. Використання даних підходів дозволяє суттєво скоротити словник слів для опрацювання шляхом приведення їх до спільної форми. Проте ситуацію ускладнює специфіка української мови, котра багата на варіації слів та їхніх відмінків. Набір інструментів для опрацювання української мови є небагатим, а існуючим інструментам може бракувати точності. Це може негативно вплинути на кінцевий результат. Алгоритми стемінгу можуть змінювати слова занадто (Over-Stemming) чи недостатньо (Under-Stemming) [15]. Для приклада наведемо різницю між результатами стемінгу двох алгоритмів із різними показниками OI (overstemming index) та UI (understemming index) [16] на основі варіацій слова зберегти (табл. 1).

Таким чином слід ретельно підбирати алгоритми стемінгу та лематизації твітів. Для використання в наступних етапах обрані інструменти tree_stem [16] для стемінгу та для лематизації – simplelemma [17], що ґрунтується на алгоритмах лексичного аналізу, описаних в [18], [19] та [20]. Відслідкуємо вплив на опрацювання твітів робить стемінг та лематизація на прикладі вибірки в 9000 твітів (табл. 2).

Таблиця 2 – Дослідження впливу стемінгу та лематизації на вибірку

Метод	Словник	Час виконання (мс)
Оригінал	15962	–
Стемінг	9543	1292
Лематизація	10770	647

Згідно з результатами, обидва підходи скорочують словник приблизно на третину. Стемер скорочує словник дещо більше, проте використовує у два рази більше часу. Оскільки лематизація приводить слово до його основної форми, а не скорочує його, після такого опрацювання текст зберігає більше семантичної повноти. Після проведення попереднього опрацювання тексту відбувається векторизація та кластеризація тексту. Враховуючи особливості української мови та інструментів її опрацювання, неможливо точно визначити який алгоритм кластеризації та векторизації дасть більш оптимальний результат. Порівняння підходів здійснюватиметься на основі вибірки в 5000 твітів. Розпочнем із алгоритму кластеризації K-Means (рис. 2, табл. 3). Для початка побудуємо стовпчикові діаграми кластерів, де вісь X позначає номер кластеру, а вісь Y – розмір кластеру. Для кожного варіанту створюємо 100 кластерів. Для оцінки результатів кластеризації введемо низку параметрів:

1) t – час виконання (сек.).

2) σ (%) – відношення стандартного відхилення до середнього значення. Дозволяє оцінити рівномірність розподілу кластерів.

Таблиця 3 – Результати кластеризації варіацій підходів для k-means

Варіант	*t*	σ	*sc*	*nc*	*n*
original + TF-IDF	9,170	246	0,051	2	0,08
stemming + TF-IDF	6,246	223	0,058	1	22,8
lemmatizing + TF-IDF	8,246	405	0,026	66	84,4
original + BOW	11,308	426	–0,028	62	57,7
stemming + BOW	8,558	761	0,008	67	79,5
lemmatizing + BOW	9,551	612	–0,026	59	63,88
original + BERT	438,03	85	0,033	6	0,30
stemming + BERT	385,74	84	0,040	4	0,24
lemmatizing + BERT	462,37	84	0,038	6	0,42

Таблиця 1 – Порівняння стемінгу різних інструментів

Алгоритм	Слова						
Оригінал	зберігайте	зберігай	зберіг	зберігав	зберігати	зберігала	зберігатимемо
urk_stemmer	зберігайт	зберіга	зберіг	зберігал	зберіг	зберіг	зберігатимем
tree_stem	зберіга	зберіга	збер	зберіга	зберіга	зберіга	зберіга

DOI 10.15588/1607-3274-2023-2-11

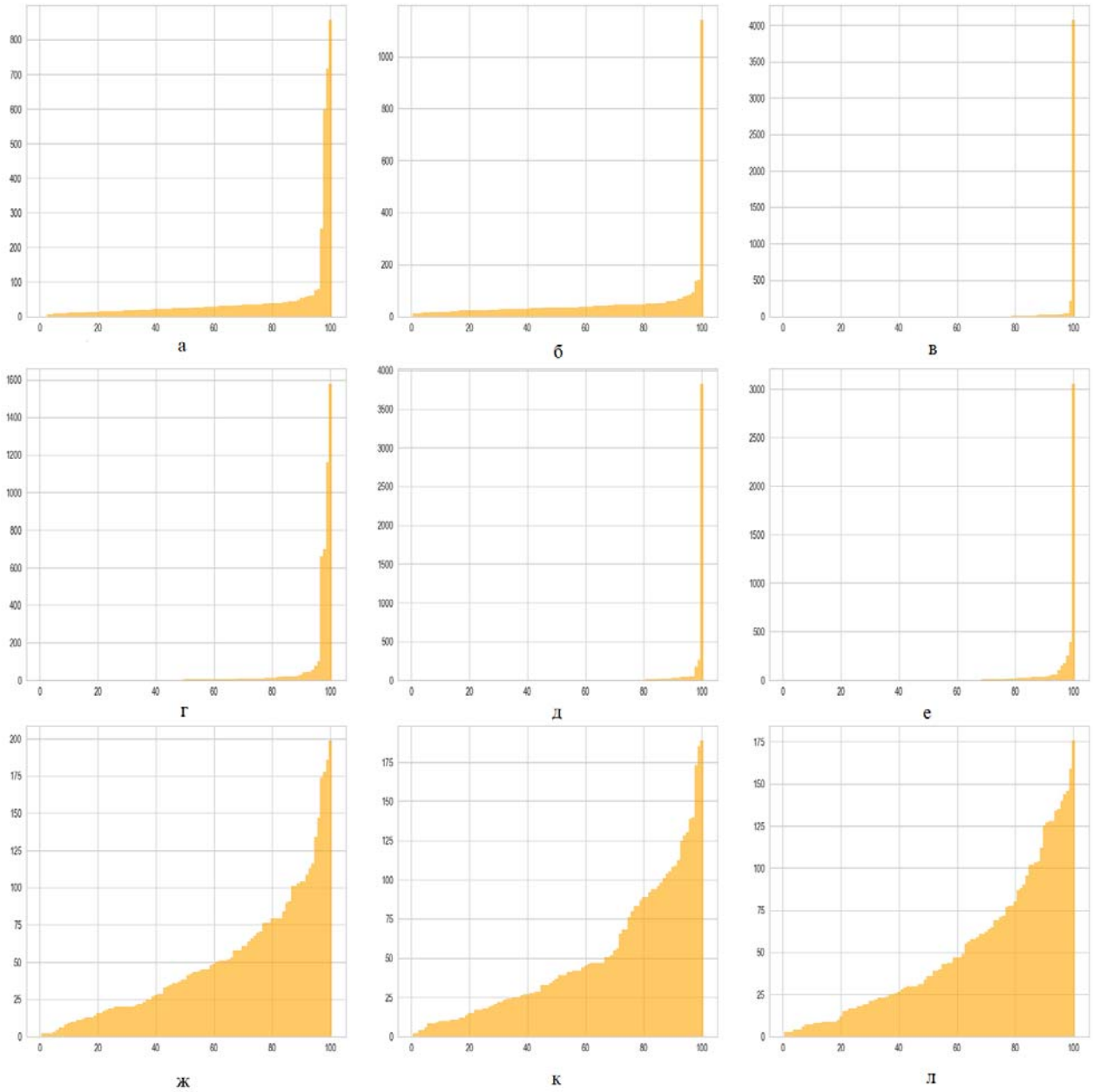

Рисунок 2 – Стовпчикові діаграми варіацій підходів для K-Means при застовуванні варіантів original + TF-IDF (а), stemming + TF-IDF (б), lemmatizing + TF-IDF (в), original + BOW (г), stemming + BOW (д), lemmatizing + BOW (е), original + BERT (ж), stemming + BERT (к) та lemmatizing + BERT (л)

3) Silhouette coefficient (*sc*) – силуетна оцінка кластерів, що показує наскільки кластери розподілені між собою. Знаходиться в межах [–1; 1], де –1 означає, що кластери розподілені неправильно, 1 означає, що кластери добре розподілені, а 0 – означає, що відстань між кластерами не є достатньо значною, щоб оцінити розподіл кластерів.

4) Noise clusters % (*nc*) – відношення кількості кластерів, що є непридатними для подальшого використання, до загальної кількості кластерів. Це є кластери, що містять замало твітів (менше 5) або кластери що містять занадто багато повідомлень, щоб відображати лише 1 тему (більше 20% вибірки).

5) Noise % (*n*) – відношення кількості повідомлень шумових кластерів до розміру вибірки.

Результати показують, що використання векторизації BOW програє при будь-яких варіантах. Хоч час виконання невеликий, проте такі кластери містять дуже велике відхилення, значну кількість шуму та погану силуетну оцінку.

Використання TF-IDF векторизації дає кращі результати та невелику кількість шуму за короткий час. Також цей векторизатор дає найкращу силуетну оцінку, проте процент відхилення все ще залишається великим. Стемінг дещо покращив результат кластеризації, на відміну від лемматизації.

BERT векторизація забезпечує найкращий сумарний результат, створюючи найменше

DOI 10.15588/1607-3274-2023-2-11

відхилення, дуже низький рівень шуму та середню силуетну оцінку, проте час виконання таких варіантів перевищує час виконання попередніх в десятки разів, що може стати проблемою при опрацюванні більших вибірок твітів в тисячі чи сотні тисяч твітів.

Стемінг у більшості випадків покращує результат кластеризації, скорочуючи її час, що стає в нагоді для BERT, також зменшує відхилення та покращує силуетну оцінку. Лемматизація навпаки збільшує час, шум та відхилення при дослідженні україномовних твітів.

Оскільки наступні алгоритми кластеризації, що будуть досліджуватись визначають кількість кластерів динамічно, додаєм до результатів ще один показник (k), що означає кількість кластерів.

Цей набір показує досить цікаві результати (рис. 3, табл. 4). Агломеративна кластеризація займає більше часу, ніж к-середніх, при цьому застосування TD-IDF векторизації дає зовсім посередні результати. Проте із BOW та BERT векторами ситуація докорінно змінюється. Відхилення та силуетна оцінка набувають дуже позитивних значень, але при цьому кількість кластерів досягає понад 2500, а шум до 99%. Майже всі кластери містять лише декілька твітів, що не дивно при такій кількості кластерів. З іншого боку при використанні інакших вхідних параметрів можна досягнути низької кількості кластерів із низьким рівнем шуму. Стемінг та лематизація не здійснюють особливого впливу на якість кластеризації, проте можуть суттєво скоротити час опрацювання.

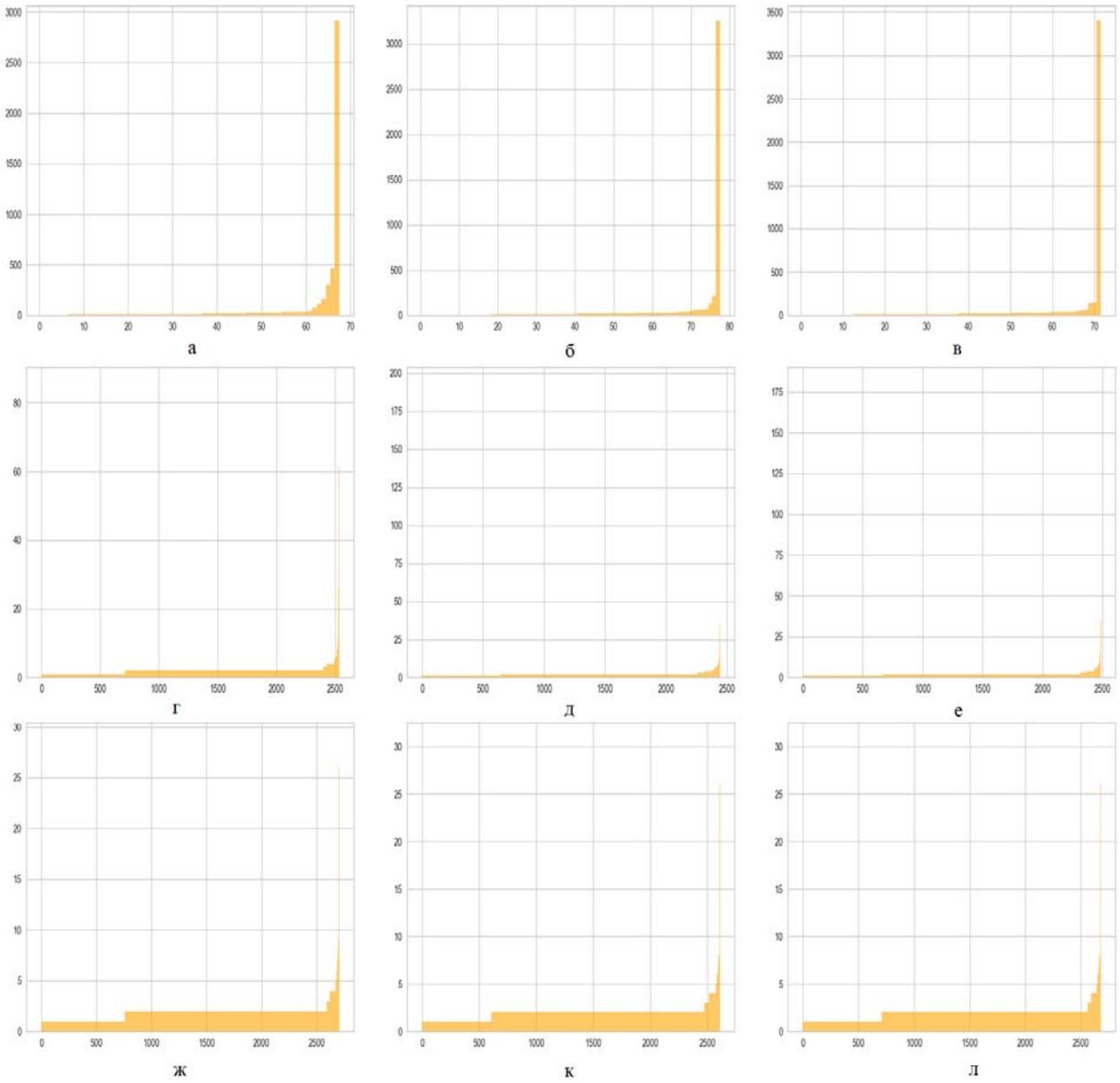

Рисунок 3 – Стовпчикові діаграми варіацій підходів для Agglomerative clustering при застовуванні варіантів original + TF-IDF (а), stemming + TF-IDF (б), lemmatizing + TF-IDF (в), original + BOW (г), stemming + BOW (д), lemmatizing + BOW (е), original + BERT (ж), stemming + BERT (к) та lemmatizing + BERT (л)

DOI 10.15588/1607-3274-2023-2-11

Таблиця 4 – Результати кластеризації варіацій підходів для Agglomerative

Варіант	*k*	*t*	σ	*sc*	*nc*	*n*
original + TF-IDF	67	169,13	477	0,04	1	58.3
stemming + TF-IDF	77	93,66	565	0,05	1	65
lemmatizing + TF-IDF	71	107,17	566	0,05	1	68
original + BOW	2535	162,77	136	0,076	98	88.4
stemming + BOW	2442	97,14	202	0,75	97	85.9
lemmatizing + BOW	2485	109,02	190	0,75	98	87.5
original + BERT	2707	483,96	59	0,80	99	94
stemming + BERT	2611	472,98	58	0,80	99	94
lemmatizing + BERT	2677	450,55	60	0,80	99	94

Алгоритм HDBSCAN також визначає кількість кластерів динамічно, а також він визначає шум власноруч (рис. 4, табл. 5). Це означає, що в результаті кластеризації немає шумних кластерів, оскільки усі шумні повідомлення відсіюються в процесі, тому із результату зникає параметр nc. Оскільки алгоритм власноруч визначає шум, замість силуентної оцінки для визначення розподіленості кластерів використовуємо оцінку DBCV [21], що враховує шум, а також для виміру використовує не відстань, а щільність. Дана оцінка також лежить в межах [–1; 1].

Поєднання NLP-методів разом із HDBSCAN дають позитивні результати. Кількість кластерів для вибірки в 5000 твітів знаходиться в межах від 50 до 200, відхилення не перевищує 200%, та *dbcv* оцінка не набуває негативних значень. Проте варіанти із HDBSCAN є найтривалішими в опрцюванні, а також відсіюють багато шуму.

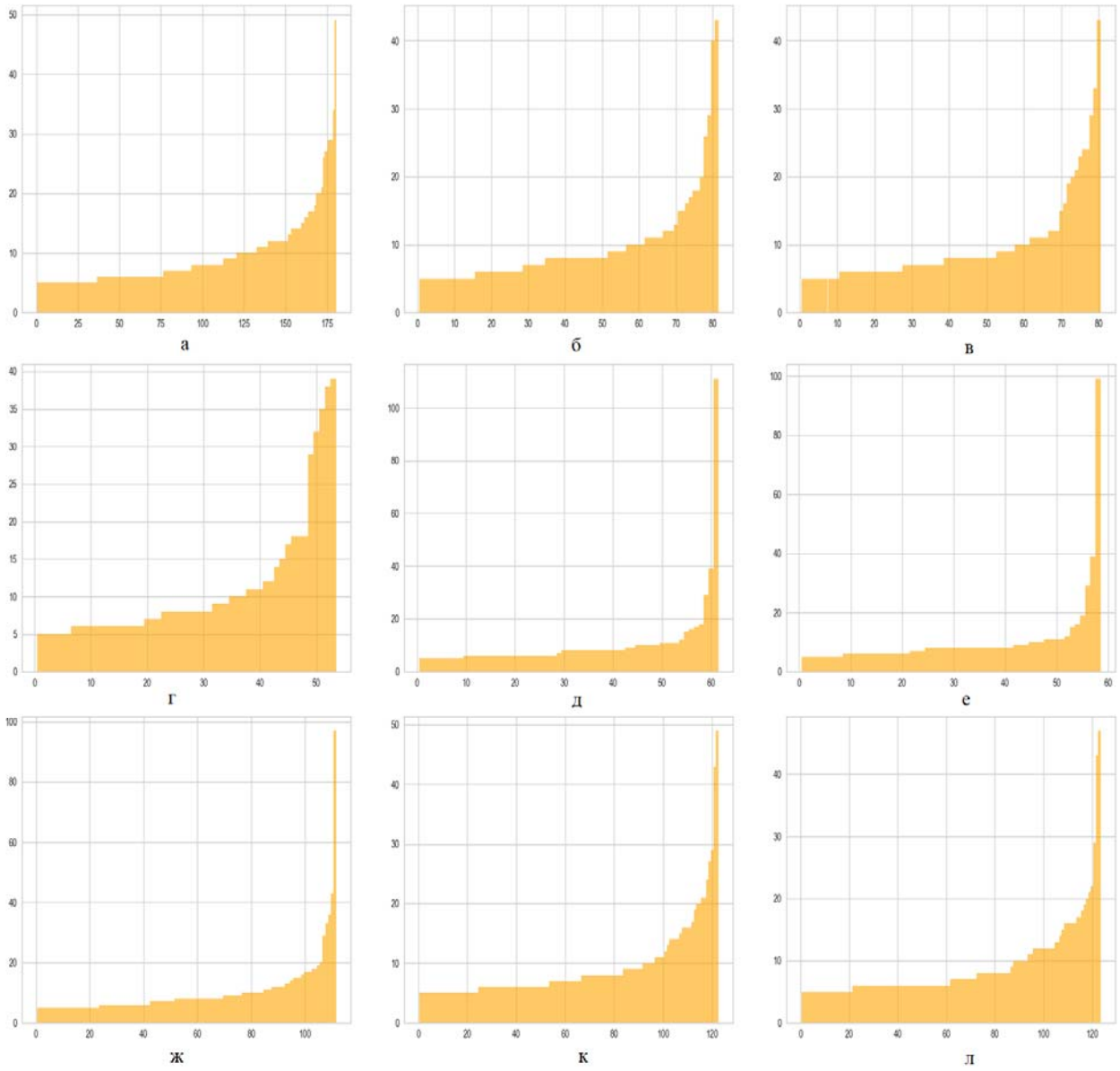

Рисунок 4 – Стовпчикові діаграми варіацій підходів для HDBSCAN при застовуванні варіантів original + TF-IDF (а), stemming + TF-IDF (б), lemmatizing + TF-IDF (в), original + BOW (г), stemming + BOW (д), lemmatizing + BOW (е), original + BERT (ж), stemming + BERT (к) та lemmatizing + BERT (л)

DOI 10.15588/1607-3274-2023-2-11

Таблиця 5 – Результати кластеризації варіацій підходів для HDBSCAN

Варіант	*k*	*t*	σ	*dbcv*	*n*
original + TF-IDF	180	471,10	65	0,050	66,1
stemming + TF-IDF	81	378,89	70	0,027	84,3
lemmatizing + TF-IDF	80	461,87	68	0,028	84,3
original + BOW	53	599,43	76	0,030	88,2
stemming + BOW	61	378,09	134	0,034	87,2
lemmatizing + BOW	58	487,75	124	0,035	87,9
original + BERT	111	477,76	101	0,074	77,2
stemming + BERT	122	442,50	72	0,068	77,3
lemmatizing + BERT	123	486,91	69	0,063	77,6

Згідно з результатами проведених кластеризацій, оптимальними рішеннями є використання алгоритму кластеризації HDBSCAN та алгоритму векторизації BERT для досягнення найточніших результатів, та використання K-Means разом із TF-IDF для досягнення найкращої швидкодії із оптимальним результатом. Для зменшення часу виконання можна застосовувати стемінг.

5 РЕЗУЛЬТАТИ

Після етапу кластеризації твітів за день йде етап злиття із кластерами попереднього дня. Злиття проводиться лише між кластерами із подібними відбитками, для цього необхідно знайти шлях перевіряти подібність відбитків. Розглянемо низку алгоритмів пошуку подібності тексту у порівнянні кластерів. Створимо два набори кластерів з 9000 твітів за один день, кожен набір міститиме 4500 твітів взятих із вибірки у шаховому порядку. Таким чином можна створити набори кластерів, що точно будуть містити подібні елементи. Із кожного набору створимо по 100 кластерів та порівняємо їх між собою. Це 10000 перевірок з яких до 100 з них мають визначити високу подібність, а решта – низьку.

Підготуємо гістограми із результатами перевірок алгоритмів, де на осі *X* розміщені подібності, а на осі *Y* – їхня частота при перевірках (рис. 5). Окрім цього із двох наборів було вручну обрано два кластери, відбитки яких повинні мати велику схожість. Таким чином можна перевірити відмінність між результатом перевірки схожих кластерів та середніми результатами.

Для порівняння скористаємось такими метриками:

1) t – час порівняння;
2) s_{avg} – середня подібність;
3) s_1 – подібність кластерів, вибраних вручну;
4) s_Δ – різниця подібностей.

Майже всі алгоритми здійснили 10000 перевірок досить швидко, за винятком Sift4 (рис. 5, табл. 6). Час виконання відіграє важливу роль, коли потрібно здійснювати тони таких перевірок. Певна частина алгоритмів має дуже низьку різницю між середньою подібністю та високою подібністю (<20). Такі алгоритми не є оптимальним вибором для порівнянь відбитків кластерів, оскільки для них складніше вибрати межу, за якою кластер вважається подібним. Найкращі результати показали алгоритми, що використовують Відстань Левенштейна, тобто *fuzz sort*, *fuzz set* та *levenshtein*. Дані алгоритми швидко здійснюють перевірки, мають більшу різницю подібностей, тож можна точніше визначити межу подібності.

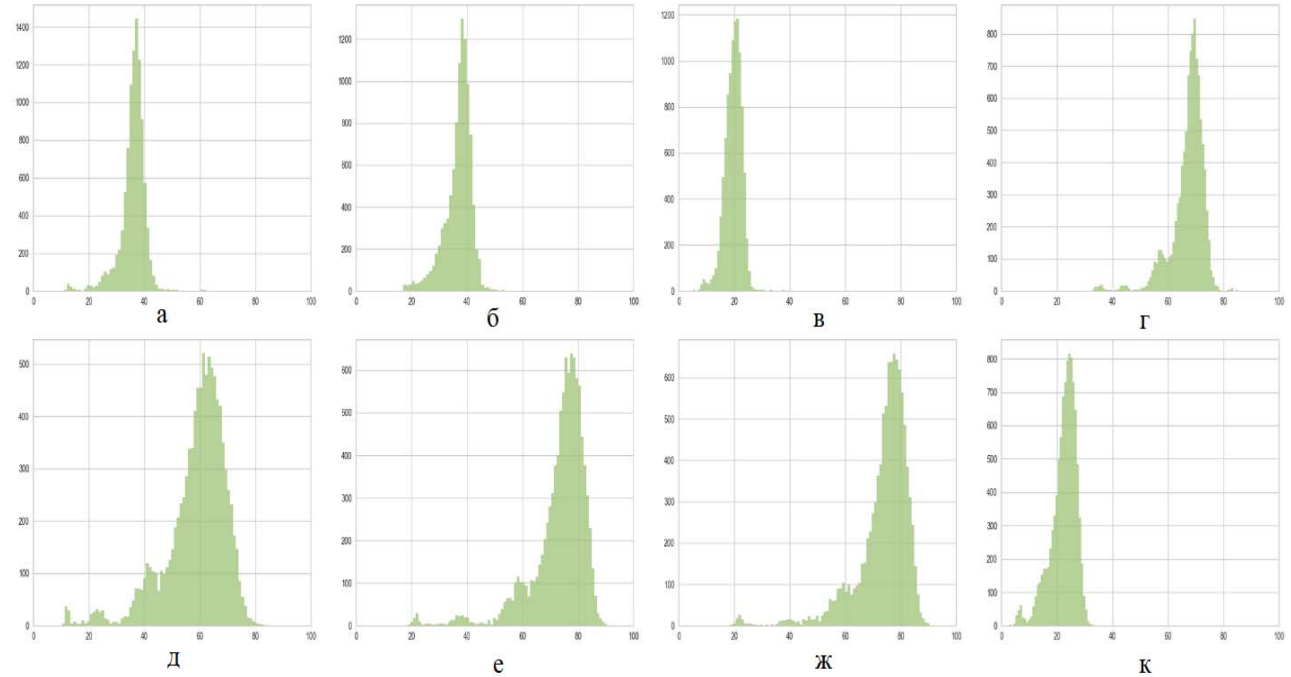

Рисунок 5 – Гістограми подібностей перевірок кластерів різними алгоритмами як Fuzz sort (а), Fuzz set (б), Levenshtein (в), Jaro Winkler (г), Jaccard (д), Sorensen (е), Cosine (ж) та Sift4 (к)

DOI 10.15588/1607-3274-2023-2-11

Таблиця 6 – Результати порівняння алгоритмів визначення подібності тексту

Алгоритм	t, сек.	s_{avg}, %	s_1, %	s_Δ, %
Fuzz sort	1,79	35,72	68	32,28
Fuzz set	1,70	36,71	97	60,29
Levenshtein	0,42	19,62	48,05	28,43
Jaro Winkler	0,31	67,15	86,66	19,51
Jaccard	1,25	58,48	75,37	16,89
Sorensen	1,10	73,07	85,96	12,89
Cosine	1,09	73,56	86	12,44
Sift4	25,44	22,40	42,74	20,34

6 ОБГОВОРЕННЯ

Здійснення вибору основних алгоритмів, необхідних в пайплайні, дозволяє спробувати провести процес пайплайну для декількох днів та отримати певні результати. Для цього здійснимо вибірку твітів із семи днів тижня, з кожного дня по 9000 твітів, здійснимо кластеризацію, злиття, та зобразимо популярність обраних кластерів на графіку (рис. 6).

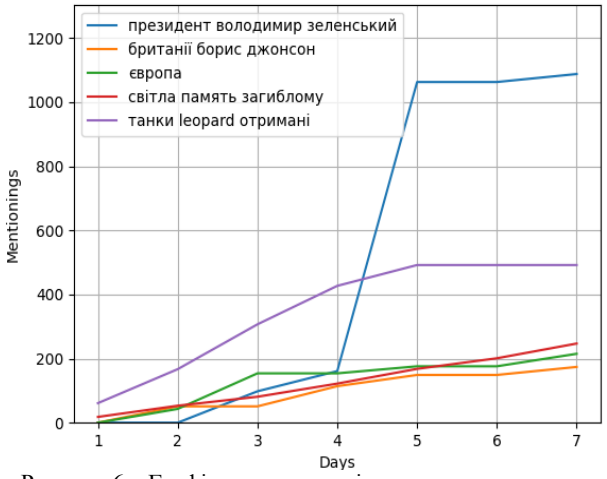

Рисунок 6 – Графік популярності тем протягом тижня

Даний графік отриманий за допомогою K-Means та TF-IDF для кластеризації та векторизації, а також fuzz sort для порівняння відбитків кластерів із межою подібності 55%.

Як бачимо з графіку для лютого 2023 року для українців залишаються самими актуальним темами твіти з ключовими словами президент Володимир Зеленський (блакитний) та танки leopard отримані (фіолетовий). На другому місці популярності є теми світла пам'ять загиблому (червона), Європа (зелена) та Борис Джонсон (оранжевий).

Реалізуємо пайплайн також й для інших алгоритмів кластеризації. Agglomerative кластеризація відзначилась нестабільністю, проте досить збалансованих результатів вдалось досягти у поєднанні із векторизацією TF-IDF. Проведемо пайплайн для цієї комбінації (рис. 7).

Набір тем для цього варіанту є подібним до попереднього за виключенням теми Європа (відбулась заміна на тематику – зменшилося виробництво електроенергії – зелена). Розподіл згадувань також є подібним, проте в даному варіанті він є більш рівномірним. Зрештою, ціною незначного

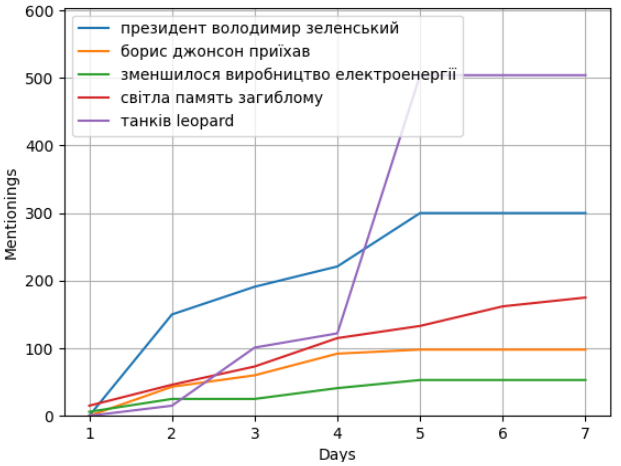

Рисунок 7 – Графік популярності тем протягом тижня для Agglomerative

покращення рівномірності кластерів є значне збільшення тривалості виконання.

Останньою комбінацією для проведення аналізу є HDBSCAN + BERT (рис. 8).

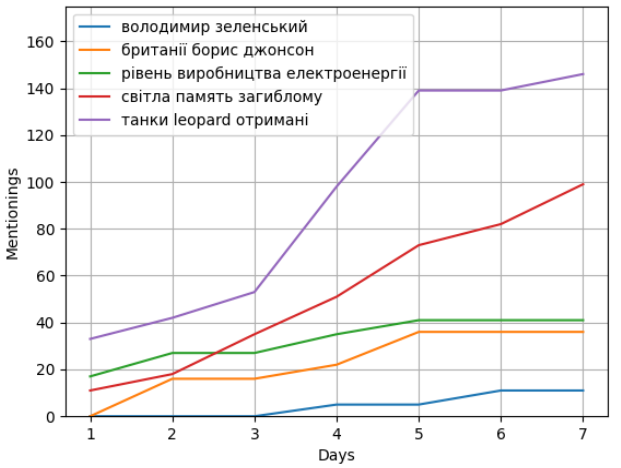

Рисунок 8 – Графік популярності тем протягом тижня для HDBSCAN

Це дослідження показало найкращі метрики якості кластерів при їхньому аналізі (табл. 5). Розподіл тем за такої комбінації практично збігається із результатом для Agglomerative (рис. 7). Із отриманого графіку видно, що загальна кількість згадувань кожної із тем невелика. Це результат визначення значної кількості твітів, як шуму. Час виконання аналізу в декілька разів перевищує Agglomerative кластеризацію, що робить цей підхід непридатним для щоденного аналізу значної кількості твітів (> 100 000). Отримані результати містять як передбачувані, так і досить неочікувані висновки. Невисока точність певних алгоритмів стемігну може бути пов’язана із різноманіттям української мови так, що алгоритми не завжди приводять до основи коректно. Незважаючи на це, стемінг все ще здійснює позитивний вплив на кінцеву вибірку. Сказати чому лематизація не дає такого ж ефекту, хоча вона також зменшує вхідний словник слів, досить складно. Це може бути пов’язане

DOI 10.15588/1607-3274-2023-2-11

із точністю обраного інструменту або із особливостями алгоритмів векторизації та кластеризації, оскільки при певних комбінаціях покращення, все ж, спостерігалось.

Алгоритм векторизації тексту TF-IDF має перевагу над Bag of Words за рахунок врахування частоти слів не тільки в окремих твітах, але й в усій вибірці загалом. Усі алгоритми кластеризації показали цікаві результати. Нестабільність числа кластерів в агломератовій кластеризації може бути спричинене чутливістю алгоритму до вхідних даних та параметрів. HDBSCAN будує хороші кластери за рахунок підходу, що базується на щільності, проте, через різноманітність твітів, щільність є часто невеликою, що залишає, в решті-решт, багато шуму.

Для порівняння відбитків найбільш оптимальними методами стали fuzz sort, fuzz set та levenshtein. Дані методи використовують відстань Левенштейна у своїй роботі. Це показує, що такий підхід добре працює для порівняння наборів ключових слів. Інші алгоритми не можуть показати велику відмінність між різними відбитками, що свідчить про те, що вони не адаптовані до такого роду задач.

Зрештою, отримана статистика популярності кластерів за сім днів. На відміну від статистики, побудованої лише по ключових словах, що відображається, як один із результатів дослідження твітів щодо COVID-19 [22], в запропонованому підході одиницею виміру є не слово чи фраза, а кластер, відбиток якого зберігається для аналізу, або злиття із іншими кластерами. Аналіз відбитків дозволить отримувати результати, як, наприклад, емоційний аналіз, що зображено в тому ж дослідженні або у ще одному дослідженні щодо COVID-19 [23], проте підхід кластеризації та злиття відбитків дозволяє розширювати дану статистику щоденно в автоматичному режимі. Кластеризація твітів вже описана в дослідженні алгоритму DBSCAN для побудови трендів твітів [24]. Проведене дослідження пропонує порівняння різних підходів до кластеризації чи векторизації твітів, а також дозволяє будувати статистику ітеративно. Це дозволяє оптимізувати процес аналізу емоційного стану соціально-активгної спільноти для формування пронозів щодо динаміки зміни громадської думки в конретному тематичному інформаційногму просторі [25–27]. На останок, дане дослідження здійснене на основі аналізу україномовних твітів та визначає оптимальний набір інструментів для роботи із українською мовою.

ВИСНОВКИ

Вирішено актуальне завдання в моделюванні процесів дослідження громадської думки на основі інтелектуального аналізу україномовного потоку контенту в соціальних мережах.

Наукова новизна отриманих результатів полягає у тому, що вперше запропоновано метод знаходження кластеризованих тематичних груп україномовних твітів на різних рівнях попереднього опрацювання україномовних текстів для визначення ефективності застосування певних NLP підходів, алгоритмів чи інструментів, що дозволило покращити точність пошуку поідбного контенту. Дослідження здійснене на основі аналізу україномовних твітів та визначає оптимальний набір інструментів для роботи із українською мовою. Досліджено особливості збору, фільтрації та попередньої обробки твітів із врахуванням особливостей української мови. Серед них такі, як: здійснення вибірки твітів зважаючи на подібність української та російської мов, а також імовірність присутності інших мов в україномовних твітах; Видалення зайвих елементів із твітів; Вибір оптимальних інструментів для стемінгу та лематизації українського тексту, зважаючи на відсутність широкого вибору, та дослідження впливу такої обробки тексту на вхідний словник слів. Організовано послідовність процесів у пайплайн, щоденне виконання якого дозволяє будувати тренди тем україномовних твітів, а кожна з тем містить відбиток, який можна використовувати для подальшого аналізу, як частину дослідження громадської думки.

Практична цінність полягає у тому, що вона є структурною складовою процесу оцінювання подібності текстового контенту на основі попереднього опрацювання тексту, векторизації та кластеризації. Здійснено порівняння алгоритмів кластеризації та векторизації тексту за допомогою визначених метрик. Досліджено вплив стемінгу та лематизації на час виконання загальної обробки та на якість отриманих кластерів. Отримано оптимальні варіації алгоритмів для точного та швидкого варіантів використання. Здійснено порівняння алгоритмів визначення подібності тексту для обрахування схожості відбитків кластерів. Знайдено оптимальні, для використання, алгоритми, що надають хорошу точність та продуктивність. Проведено дослідження трендів одного тижня часу та визначено зміну популярності на прикладі декількох обраних тем.

В результаті проведених експериментів реалізовано та проаналізовано функціонування пайплайн із збору та опрацювання україномовних твітів. На різних етапах опрацювання проведені експерименти для визначення ефективності застосування певних NLP підходів, алгоритмів чи інструментів. Спочатку встановлено, що стемінг та лематизація дозволяють суттєво скоротити вхідний словник слів. Обрано оптимальні алгоритми, згідно їхньої ефективності при опрацюванні української мови. Розглянуто приклад незадовільного стемінгу.

Далі проведено детальне порівняння варіацій кластеризації твітів, що полягають в поєднанні NLP-методів попереднього опрацювання тексту, векторизації та кластеризації. Здійснено 27 різних кластеризацій та проаналізовано їхні результати. Експеримент показав, що стемінг переважно скорочує час опрацювання україномовних твітів, та покращує якість кластеризації. Лематизація також пришвидшує процес, проте часто призводить до погіршення результатів. Алгоритм векторизації тексту Bag of Words програє іншим алгоритмам таким як TF-IDF та

DOI 10.15588/1607-3274-2023-2-11

BERT. TF-IDF є оптимальним вибором з точки зору швидкості роботи. Хоч BERT і призводить до покращення результатів, проте така векторизація потребує набагато більше часу. HDBSCAN кластеризація створює найбільш рівнорозподілені кластери, проте також займає багато часу. K-Means є найшвидшим алгоритмом кластеризації серед розглянутих. Агломератова кластеризація хоч і може давати дещо кращі кластери, проте результати її роботи виявились досить нестабільними.

Серед інструментів знаходження подібності тексту найкраще для порівняння відбитків кластерів підходять fuzz sort, fuzz set та levenshtein. За швидкістю здійснення перевірку найкращий результат показав метод levenshtein. Два інших втричі гірше показали швидкість перевірки, але вони майже в 13 разів швидше працюють, ніж Sift4. Найшвидший метод є Jaro Winkler, але він має 19,51% різницю подібностей. Кращу різницю подібностей має метод fuzz set (60,29 %). Друге місце посіли Fuzz sort (32,28%) та Levenshtein (28,43%). Дані методи використовують відстань Левенштейна у своїй роботі. Це показує, що такий підхід добре працює для порівняння наборів ключових слів. Інші ж інструменти показують занадто низьку різницю між середньою подібністю відбитків та подібністю вручну вибраних схожих кластерів. У деяких алгоритмів середня подібність відбитків сягає вище 70%.

В кінці проведення пайплайну отримано графік популярності декількох кластерів за 7 днів часу. Такий процес можна проводити щоденно й постійно отримувати актуальні дані.

Перспективи подальших досліджень полягають в дослідженні громадської думки на основі інших алгоритмів визначення подібності тексту для обрахування подібності відбитків кластерів.

ПОДЯКИ

Роботу виконано в рамках держбюджетної теми «Методи та засоби функціонування систем підтримки прийняття рішень на основі онтологій» (ID:839 2017-05-15 09:20:01 (2459-315)). Дослідження провадилось в межах спільних наукових досліджень кафедри інформаційних систем та мереж НУ «Львівська політехніка» на тему «Дослідження, розроблення і впровадження інтелектуальних розподілених інформаційних технологій та систем на основі ресурсів баз даних, сховищ даних, просторів даних та знань з метою прискорення процесів формування сучасного інформаційного суспільства». Наукові дослідження провадилися також в рамках ініціативної тематики досліджень кафедри ІСМ НУ «Львівська політехніка» на тему «Розроблення інтелектуальних розподілених систем на основі онтологічного підходу з метою інтеграції інформаційних ресурсів».

ЛІТЕРАТУРА

1. Ismail M. A. On Time Series Analysis for Repeated Surveys / M. A. Ismail, H. A. Auda, Y. A. Elzafrany // Journal of Statistical Theory and Applications. – 2018. – Vol. 17. – P. 587–596. https://doi.org/10.2991/jsta.2018.17.4.1
2. Mellon J. Twitter and Facebook are not representative of the general population: Political attitudes and demographics of British social media users / J. Mellon, C. Prosser // Research & Politics. – 2017. – Vol. 4(3). – P. 1–9. https://doi.org/10.1177/2053168017720008
3. Using Social Media to Mine and Analyze Public Opinion Related to COVID-19 in China / [X. Han, J. Wang, M. Zhang, X. Wang] // International Journal of Environmental Research and Public Health. – 2020. – Vol. 17(8). – P. 2788. https://doi.org/10.3390/ijerph17082788
4. Twitter as a sentinel tool to monitor public opinion on vaccination: an opinion mining analysis from September 2016 to August 2017 in Italy / [L. Tavoschi, F. Quattrone, E. D'Andrea et al.] // Human Vaccines & Immunotherapeutics. – 2020. – Vol. 16(5). – P. 1062–1069. https://doi.org/10.1080/21645515.2020.1714311
5. Sentiment analyses of multilingual tweets on halal tourism / [S. Ainin, A. Feizollah, N. B. Anuar, N. A. Abdullah] // Tourism Management Perspectives. – 2020. – Vol. 34. – P. 100658. https://doi.org/10.1016/j.tmp.2020.100658
6. Twitter Inc. Twitter API / Twitter Inc. – Access mode: https://developer.twitter.com/en/docs/twitter-api
7. Moh T.-S. Clustering of Technology Tweets and the Impact of Stop Words on Clusters / T.-S. Moh, S. Bhagvat // ACM-SE : the 50th Annual Southeast Regional Conference : Tuscaloosa, Alabama, 29–31 March 2012 : proceedings. – Alabama: ACM-SE, 2012. – P. 226–231. https://doi.org/10.1145/2184512.2184566
8. Mitsch R. SpaCy. Explosion. Industrial-strength Natural Language Processing (NLP) in Python / R. Mitsch. – Access mode: https://github.com/explosion/spaCy
9. Kupriienko S. Ukrainian-Stopwords / S. Kupriienko. – Access mode: https://github.com/skupriienko/Ukrainian-Stopwords
10. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding / [J. Devlin, M.-W. Chang, K. Lee, K. Toutanova] // ArXiv. – 2018. https://doi.org/10.48550/arXiv.1810.04805
11. BERT multilingual base model (uncased) / F. Hugging. – Access mode: https://huggingface.co/bert-base-multilingual-uncased
12. Sasirekha K. Agglomerative hierarchical clustering algorithm-a / K. Sasirekha, P. Baby // International Journal of Scientific and Research Publications. – 2013. – Vol. 83(3). – P. 83.
13. McInnes L. Accelerated Hierarchical Density Based Clustering / L. McInnes, J. Healy // Data Mining Workshops (ICDMW) : International Conference, New Orleans, LA, USA, 18 December 2017 : proceedings. – New Orleans: IEEE, 2017. – P. 33–42. https://doi.org/10.1109/ICDMW.2017.12
14. YAKE! Keyword extraction from single documents using multiple local features / [R. Campos, V. Mangaravite, A. Pasquali et al.] // Information Sciences. – 2020. – Vol. 509. – P. 257–289. https://doi.org/10.1016/j.ins.2019.09.013
15. Paice C. D. An Evaluation Method for Stemming Algorithms / C. D. Paice // Research and Development in Information Retrieval : the 17th Annual International ACM SIGIR Conference, Dublin, Ireland, 1994 : proceedings. – Berlin, Heidelberg: Springer-Verlag, 1994. – P. 42–50.
16. Makukha A. Stemmers for Ukrainian / A. Makukha. Access mode: https://github.com/amakukha/stemmers_ukrainian

DOI 10.15588/1607-3274-2023-2-11

17. Barbaresi A. Simplemma / A. Barbaresi // Zenodo. – 2023. https://doi.org/10.5281/zenodo.7555188
18. Barbaresi A. Data-Driven Identification of German Phrasal Compounds / A. Barbaresi, K. Hein // Lecture Notes in Computer Science. – 2017. – Vol. 10415 – P. 192–200. https://doi.org/10.1007/978-3-319-64206-2_22
19. Barbaresi A. An Unsupervised Morphological Criterion for Discriminating Similar Languages / A. Barbaresi // NLP for Similar Languages, Varieties and Dialects (VArDIal3) : Third Workshop, Osaka, Japan, December 2016 : proceedings. – Osaka: ACL Anthology, 2016. – P. 212–220.
20. Barbaresi A. Bootstrapped OCR error detection for a less-resourced language variant / A. Barbaresi // Natural Language Processing (KONVENS) : 13th Conference, Bochum, Germany, September 2016 : proceedings. – Berlin: HAL,2016. – P. 21–26.
21. Density-Based Clustering Validation / [D. Moulavi, P. A. Jaskowiak, R. J. G. B. Campello et al.] // Data Mining (SDM) : the 2014 SIAM international conference, Philadelphia, Pennsylvania, USA, 24–26 April 2014: proccedings. – Philadelphia: SIAM, 2014. – P. 839–847. https://doi.org/10.1137/1.9781611973440.96
22. Boon-Itt S. Public Perception of the COVID-19 Pandemic on Twitter: Sentiment Analysis and Topic Modeling Study / S. Boon-Itt, Y. Skunkan // JMIR Public Health Surveill. – 2020. – Vol. 6(4). – P. e21978. https://doi.org/10.2196/21978
23. Global Sentiments Surrounding the COVID-19 Pandemic on Twitter: Analysis of Twitter Trends / [M. O. Lwin, J. Lu, A. Sheldenkar et al.] // JMIR Public Health Surveill. – 2020. – Vol. 6(2). – P. e19447. https://doi.org/10.2196/19447
24. DBSCAN algorithm: twitter text clustering of trend topic pilkada pekanbaru / [Mustakim, R. N. G. Indah, R. Novita, et al.] // Journal of Physics. – 2019. – Vol. 1363(1). – P. 012001. https://doi.org/10.1088/1742-6596/1363/1/012001
25. Emotion recognition system project of English newspapers to regional E-business adaptation / [O. Markiv, V. Vysotska, L. Chyrun et al.] // Computer science and information technologies : IEEE 17th International conference, Lviv, Ukraine, 10–12 November 2022 : proceedings. – Lviv: IEEE, 2022. – P. 392–397. https://doi.org/10.1109/CSIT56902.2022.10000527
26. Sentiment analysis technology of English newspapers quotes based on neural network as public opinion influences identification tool / [V. Vysotska, O. Markiv, S. Voloshyn et al.] // Computer science and information technologies : IEEE 17th International conference, Lviv, Ukraine, 10–12 November 2022 : proceedings. – Lviv: IEEE, 2022. – P. 83–88. https://doi.org/10.1109/CSIT56902.2022.10000627
27. NLP tool for extracting relevant information from criminal reports or fakes/propaganda content / [V. Vysotska, L. Chyrun, O. Brodyak et al.] // Computer science and information technologies : IEEE 17th International conference, Lviv, Ukraine, 10–12 November 2022 : proceedings. – Lviv: IEEE, 2022. – P. 93–98. https://doi.org/10.1109/CSIT56902.2022.10000563

Accepted 28.03.2023.
Received 04.05.2023.

UDC 004.9

UKRAINIAN LANGUAGE TWEETS ANALYSIS TECHNOLOGY FOR PUBLIC OPINION DYNAMICS CHANGE PREDICTION BASED ON MACHINE LEARNING

Prokipchuk O. – PhD student of Information Systems and Networks Department, Lviv Polytechnic National University, Lviv, Ukraine.

Vysotska V. – PhD, Associate Professor of Information Systems and Networks Department, Lviv Polytechnic National University, Lviv, Ukraine.

ABSTRACT

Context. Automation of public opinion research will allow not only to reduce the amount of manual work, but also to obtain time slices of the results without additional efforts. Since direct interaction with respondents should be avoided, public opinion should be analyzed based on the sources of its free expression. Social networks are great for this role, as their people freely publish their thoughts or emotionally truthfully react to published information about certain events. Statistics show that data from social networks is not enough to obtain a full-fledged result, because a significant percentage of people do not use social networks. However, the automation of the study of even such a stratum of the population is already a good result for analyzing the dynamics of changes in public opinion in accordance with events in the country/world and, accordingly, for correcting the processes of public administration in the future.

Objective of the study is to develop a technology for analyzing the flow of Ukrainian-language content in social networks for public opinion research based on finding clustered thematic groups of tweets.

Method. The article develops a technology for finding tweet trends based on clustering, which forms a data stream in the form of short representations of clusters and their popularity for further research of public opinion. An effective approach to tweet collection, filtering, cleaning and pre-processing based on a comparative analysis of Bag of Words, TF-IDF and BERT algorithms is described. The impact of stemming and lemmatization on the quality of the obtained clusters was determined. And optimal combinations of clustering methods (K-Means, Agglomerative Hierarchical Clustering and HDBSCAN) and vectorization of tweets were found based on the analysis of 27 clusterings of one data sample. The method of presenting clusters of tweets in a short format is selected.

Results. Algorithms using the Levenstein Distance, i.e. fuzz sort, fuzz set and levenshtein, showed the best results. These algorithms quickly perform checks, have a greater difference in similarities, so it is possible to more accurately determine the limit of similarity. According to the results of the clustering, the optimal solutions are to use the HDBSCAN clustering algorithm and the BERT vectorization algorithm to achieve the most accurate results, and to use K-Means together with TF-IDF to achieve the best speed with the optimal result. Stemming can be used to reduce execution time.

Conclusions. In this study, the optimal options for comparing cluster fingerprints among the following similarity search methods were experimentally found: Fuzz Sort, Fuzz Set, Levenshtein, Jaro Winkler, Jaccard, Sorensen, Cosine, Sift4. In some algorithms, the average fingerprint similarity reaches above 70%. 3 effective tools were found to compare their similarity, as they show a sufficient difference between comparisons of similar and different clusters (> 20%). Based on the selected effective methods, trend analysis was successfully performed on 90,000 tweets over 7 days for 5 topics of the week using K-Means and TF-IDF for clustering and vectorization, as well as fuzz sort for cluster fingerprint comparison with a 55% similarity threshold.

DOI 10.15588/1607-3274-2023-2-11

KEYWORDS: tweet, Ukrainian language, public opinion; trend, clustering, stemming, lemmatization, similarity of clusters.

REFERENCES

1. Ismail M. A., Auda H. A., Elzafrany Y. A. On Time Series Analysis for Repeated Surveys, *Journal of Statistical Theory and Applications*, 2018, Vol. 17, pp. 587–596. https://doi.org/10.2991/jsta.2018.17.4.1
2. Mellon J., Prosser C. Twitter and Facebook are not representative of the general population: Political attitudes and demographics of British social media users, *Research & Politics*, 2017, Vol. 4(3), pp. 1–9. https://doi.org/10.1177/2053168017720008
3. Han X., Wang J., Zhang M., Wang X. Using Social Media to Mine and Analyze Public Opinion Related to COVID-19 in China, *International Journal of Environmental Research and Public Health*, 2020, Vol. 17(8), P. 2788. https://doi.org/10.3390/ijerph17082788
4. Tavoschi L., Quattrone F., D'Andrea E., Ducange P., Vabanesi M., Marcelloni F., Lopalco P. L. Twitter as a sentinel tool to monitor public opinion on vaccination: an opinion mining analysis from September 2016 to August 2017 in Italy, *Human Vaccines & Immunotherapeutics,* 2020, Vol. 16(5), pp. 1062–1069. https://doi.org/10.1080/21645515.2020.1714311
5. Ainin S., Feizollah A., Anuar N. B., Abdullah N. A. Sentiment analyses of multilingual tweets on halal tourism *Tourism Management Perspectives*, 2020, Vol. 34, P. 100658. https://doi.org/10.1016/j.tmp.2020.100658
6. Twitter Inc. Twitter API. Access mode: https://developer.twitter.com/en/docs/twitter-api
7. Moh T.-S., Bhagvat S. Clustering of Technology Tweets and the Impact of Stop Words on Clusters, *ACM-SE : the 50th Annual Southeast Regional Conference : Tuscaloosa, Alabama, 29–31 March 2012, proceedings*. Alabama, ACM-SE, 2012, pp. 226–231. https://doi.org/10.1145/2184512.2184566
8. Mitsch R. SpaCy. Explosion. Industrial-strength Natural Language Processing (NLP) in Python. Access mode: https://github.com/explosion/spaCy
9. Kupriienko S. Ukrainian-Stopwords. Access mode: https://github.com/skupriienko/Ukrainian-Stopwords
10. Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, *ArXiv,* 2018. https://doi.org/10.48550/arXiv.1810.04805
11. Hugging F. BERT multilingual base model (uncased). Access mode: https://huggingface.co/bert-base-multilingual-uncased
12. Sasirekha K., Baby P. Agglomerative hierarchical clustering algorithm-a, *International Journal of Scientific and Research Publications*, 2013, Vol. 83(3), P. 83.
13. McInnes L., Healy J. Accelerated Hierarchical Density Based Clustering, *Data Mining Workshops (ICDMW) : International Conference, New Orleans, LA, USA, 18 December 2017 : proceedings*. New Orleans, IEEE, 2017, pp. 33–42. https://doi.org/10.1109/ICDMW.2017.12
14. Campos R., Mangaravite V., Pasquali A., Jorge A., Nunes C., Jatowt A. YAKE! Keyword extraction from single documents using multiple local features, *Information Sciences*, 2020, Vol. 509, pp. 257–289. https://doi.org/10.1016/j.ins.2019.09.013
15. Paice C. D. An Evaluation Method for Stemming Algorithms, *Research and Development in Information Retrieval, the 17th Annual International ACM SIGIR Conference, Dublin, Ireland, 1994, proceedings*. Berlin, Heidelberg, Springer-Verlag, 1994, pp. 42–50.
16. Makukha A. Stemmers for Ukrainian. Access mode: https://github.com/amakukha/stemmers_ukrainian
17. Barbaresi A. Simplemma, *Zenodo,* 2023. https://doi.org/10.5281/zenodo.7555188
18. Barbaresi A., Hein K. Data-Driven Identification of German Phrasal Compounds, *Lecture Notes in Computer Science*, 2017, Vol. 10415, pp. 192–200. https://doi.org/10.1007/978-3-319-64206-2_22
19. Barbaresi A. An Unsupervised Morphological Criterion for Discriminating Similar Languages, *NLP for Similar Languages, Varieties and Dialects (VArDIal3) : Third Workshop, Osaka, Japan, December 2016 : proceedings*. Osaka, ACL Anthology, 2016, pp. 212–220.
20. Barbaresi A. Bootstrapped OCR error detection for a less-resourced language variant, *Natural Language Processing (KONVENS) : 13th Conference, Bochum, Germany, September 2016 : proceedings*. Berlin, HAL,2016, pp. 21–26.
21. Moulavi D., Jaskowiak P. A., Campello R. J. G. B., Zimek A., Sander J. Density-Based Clustering Validation, *Data Mining (SDM) : the 2014 SIAM international conference, Philadelphia, Pennsylvania, USA, 24–26 April 2014: proccedings*. Philadelphia: SIAM, 2014, pp. 839–847. https://doi.org/10.1137/1.9781611973440.96
22. Boon-Itt S., Skunkan Y. Public Perception of the COVID-19 Pandemic on Twitter: Sentiment Analysis and Topic Modeling Study, *JMIR Public Health Surveill,* 2020, Vol. 6(4), P. e21978. https://doi.org/10.2196/21978
23. Lwin M. O., Lu J., Sheldenkar A., Schulz P. J., Shin W., Gupta R., Yang Y. Global Sentiments Surrounding the COVID-19 Pandemic on Twitter: Analysis of Twitter Trends, *JMIR Public Health Surveill*, 2020, Vol. 6(2), P. e19447. https://doi.org/10.2196/19447
24. Mustakim, Indah R. N. G., Novita R., Kharisma O. B., Vebrianto R., Sanjaya S., Hasbullah, Andriani T., Sari W. P., Novita Y., Rahim R. DBSCAN algorithm: twitter text clustering of trend topic pilkada pekanbaru, *Journal of Physics,* 2019, Vol. 1363(1), P. 012001. https://doi.org/10.1088/1742-6596/1363/1/012001
25. Markiv O., Vysotska V., Chyrun L., Voloshyn S., Dyyak I., Panasyuk V. Emotion recognition system project of English newspapers to regional E-business adaptation, *Computer science and information technologies : IEEE 17th International conference,* Lviv, Ukraine, 10–12 November 2022 : proceedings. Lviv, IEEE, 2022, pp. 392–397. https://doi.org/10.1109/CSIT56902.2022.10000527
26. Vysotska V., Markiv O., Voloshyn S., Dyyak I., Budz I., Schuchmann V. Sentiment analysis technology of English newspapers quotes based on neural network as public opinion influences identification tool, *Computer science and information technologies, IEEE 17th International conference, Lviv, Ukraine, 10–12 November 2022 : proceedings*. Lviv: IEEE, 2022, pp. 83–88. https://doi.org/10.1109/CSIT56902.2022.10000627
27. Vysotska V., Chyrun L., Brodyak O., Mazepa S., Shakleina I., Schuchmann V. NLP tool for extracting relevant information from criminal reports or fakes/propaganda content, *Computer science and information technologies : IEEE 17th International conference, Lviv, Ukraine, 10–12 November 2022 : proceedings*. Lviv, IEEE, 2022, pp. 93–98. https://doi.org/10.1109/CSIT56902.2022.10000563

DOI 10.15588/1607-3274-2023-2-11