

Програмні аспекти виявлення академічного плагіату

Микола Пікуляк, Марія Дутчак

*Кафедра інформаційних технологій
Прикарпатський національний університет імені Василя Стефаника
Івано-Франківськ, Україна*

Анотація— Виконано порівняльний аналіз методів, що застосовуються у відомих алгоритмах перевірки текстів на предмет виявлення запозичень та обчислення відсотку унікальності тексту. Виділено переваги та недоліки їх використання для перевірки наукових робіт із різних предметних областей. Проведено дослідження на основі методу статистичного групування результатів перевірки кваліфікаційних робіт у Прикарпатському національному університеті протягом двох навчальних років з використанням програмних сервісів Unicheck та Plagiat.pl, що дозволило виявити найбільш типові способи порушення академічної доброчесності студентами та визначити подальші напрямки щодо розробки вдосконалених методів та алгоритмів автоматизованого аналізу тексту. Визначено перспективи подальших досліджень, направлених на розробку сучасних програмних рішень з метою пошуку академічного плагіату.

Ключові слова—*Академічна доброчесність, Антиплагіатні програми, Методи аналізу тексту, Алгоритми створення антиплагіатних сервісів, Статистична методологія.*

I. ВСТУП

Впровадження принципів академічної доброчесності в навчальному процесі та науковій діяльності є важливою умовою успішного розвитку вищої освіти в Україні та запорукою визнання наукових досягнень українців світовою спільнотою. З однієї сторони зрозуміло, що всі нові наукові результати повинні базуватись та відштовхуватись від вже відомих. Оскільки для того, щоб створити щось нове в науці, необхідний детальний аналіз вже досягнутих успіхів у тій чи іншій предметній області. З іншої сторони, потрібно унеможливити використання у нових дослідженнях академічного плагіату, під яким розуміють оприлюднення наукових (творчих) результатів, отриманих іншими особами, як результатів власного дослідження без зазначення авторства [1].

Особливо поширеною тенденцією маніпулювання чужими ідеями та результатами в останній час є використання «рерайту» (тобто грамотного переписування оригінальної роботи) та «копіпасти» (копіювання фрагментів із декількох джерел). Якісне використання користувачами таких інструментів забезпечує внесення структурних відмінностей в текстову частину документу, хоча сам зміст та наукова складова залишаються ті ж самі, що і в джерелі.

II. АНАЛІЗ ОСТАННІХ ДОСЛІДЖЕНЬ І ПУБЛІКАЦІЙ

З метою виявлення та запобігання академічним правопорушенням на сьогодні розроблено чимало платних та безкоштовних сервісів (антиплагіатних програм), які дозволяють автоматично відстежувати текстові запозичення та визначати співпадіння й відсоток унікальності тексту в мережі Інтернет і Репозитарії. В основі їх розробки, як правило, лежать власні алгоритми аналізу текстів та певні методи пошуку прихованого плагіату, найбільш поширеними з яких є «синтаксичні» та «лексичні» методи.

До «синтаксичних» відносяться деякі прості методи синтаксичного аналізу документів, метод шинглів (алгоритм А. Бродера), Long Sent та Heavy Sent [2]. Такі методи використовують сучасні криптографічні протоколи, математичні обрахунки у яких виконують відповідні функції хешування. При цьому під хешуванням розуміють

відображення, що перетворює повідомлення довільної довжини у хеш-код фіксованої довжини n , тобто:

$$F : \{0,1\}^* \rightarrow \{0,1\}^n$$

Зокрема, у нескладних синтаксичних методах, як правило, обчислюють хеш-функцію від конкатенації двох найдовших речень, знайдених у тексті та обраховують відношення числа входжень певного слова до усієї кількості слів у тексті. Такі підходи є не точними та можуть бути використані тільки для попередньої оцінки.

У методі «шинглів» використовують послідовності фіксованої довжини, які складаються з сусідніх слів («шингли»). Від кожної послідовності («шингла») береться хеш-сума і створюється набір хеш-сум. Відсоток схожості у двох документах буде тим більшим, чим частіше перетинаються їхні множини шинглів. У кожній множині обчислюються дактилограми (тобто текстові підрядки фіксованої довжини) за допомогою взаємно-однозначних і незалежних функцій, що використовують випадкові набори простих поліномів. Головним недоліком алгоритму шинглів є зменшення продуктивності під час перевірки текстів великих об'ємів, оскільки збільшення кількості шинглів призводить до зростання операцій для виконання відповідних порівнянь.

У зв'язку з тим, що користувачі постійно знаходять нові способи обходу дії антиплагіатних програм, тому алгоритми пошуку репрезентативних підмножин постійно вдосконалюються. Сьогодні алгоритм А. Бродера отримав такі вдосконалення:

- алгоритм Карпа-Рабіна – вибирає тільки ті шингли, кількість «дактилограм» яких ділиться без залишку на деяке число m . Це дозволяє збільшити продуктивність при обробці великих обсягів тексту. Основний недолік – залежність вибірки від довжини документа;

- метод Long Sent, у якому документ спочатку розбивається на речення, які упорядковуються за зменшенням довжини, представленої кількістю слів (при рівності довжин – у алфавітному порядку). Потім вибираються та зчіплюються у рядок в алфавітному порядку два найдовші речення. В якості сигнатури документу обчислюється контрольна сума отриманого рядка. Головний недолік методу – невисока швидкість та низький процент плагіату при перевірці великого тексту;

- метод Heavy Sent – ідея методу схожа до методу Long Sent: документ ділиться на речення і два з них зчіплюють в рядок в алфавітному порядку. Однак вибір здійснюється дещо інакше: враховують вагу кожної пропозиції, що являє собою суму ваг слів, які входять до нього, і вибір припадає на дві «найважчі» пропозиції. Контрольна сума, отримана після зчеплення рядків приймається за сигнатуру документа;

- метод Moodle Crot [3] – спочатку із тексту видаляються слова до трьох символів, а потім всі небуквені знаки (дефіси, крапки, пробіли і т.д.). Потім отриманий у такий спосіб суцільний ланцюжок букв з певним кроком n «нарізається» на частини по N символів у кожній та від кожної частини береться хеш-функція, результат якої використовується для аналізу.

Прикладами лексичних методів є метод TF-IDF та Метод Коудури [4]. Основна ідея використання таких методів полягає у використанні для визначення унікальності документу спеціальних словників.

Зокрема у методі TF-IDF складається словник, що ставить кожному слову число документів, в яких воно зустрічається. Потім будується частотний словник документа і для кожного слова обчислюється його «вага». Недоліки методу – в сигнатуру не включаються рідкісні слова та низький процент плагіату при перевірці великого тексту.

У методі Коудури лексичний словник будується на основі дактилограм I-Matcha [5], за числовими значеннями яких обраховують унікальність відповідних документів. Перевагою цього методу є більш висока обчислювальна ефективність ніж в алгоритмі А. Бродера. Недоліком є нестійкість до незначних змін змісту документу. Особливо корисним його застосування є під час вебпошуку та фільтрації спаму.

Серед сучасних методів машинного навчання слід виділити метод з використанням граф-трансформерів [6], перевагою якого є можливість за рахунок так званого механізму уваги виконувати обробку будь-якого положення слова в послідовності входу та визначати міжмовні текстові збіги. Механізм уваги дозволяє у моделі декодування використовувати вектор стану кожного вхідного англomовного слова, а не лише крайнього (як це робиться в класичній моделі рекурентних нейронних мереж). Це забезпечує більш ефективний спосіб пошуку значення україномовного слова в кожному з векторів стану англomовного входу.

Також сьогодні ведуться спроби розробки програм штучного інтелекту, які на основі методів нейронних мереж та генетичних алгоритмів дозволяють проводити аналіз документів не тільки на предмет порівняння текстової частини, але й визначають плагіат зображень, моделей, конструкцій та програмних кодів.

III. МЕТОДИ

В роботі використана статистична методологія аналізу результатів дослідження, яка дозволяє на основі математичної обробки отриманих результатів оцінити динаміку зміни якості кваліфікаційних робіт, визначити найбільш типові академічні правопорушення студентами та провести статистичне узагальнення щодо розробки вдосконалених методів та алгоритмів створення антиплагіатних програм.

Перевагою застосування даної методики є можливість обчислити узагальнюючі показники зміни діапазону коефіцієнтів подібності, що описують дані окремих навчальних підрозділів та здійснити аналіз і прогнозування використаних алгоритмів аналізу текстових збігів.

Зокрема в дослідженні застосовується метод статистичного групування [7], який дає змогу виявити об'єктивний стан предметної області, простежити перехід кількісних показників у якісний аналіз текстової інформації, отримати дані про зв'язки між досліджуваними показниками, їх співвідношення в загальній сукупності та оцінити закономірності їхнього розвитку.

Методологічна сутність запропонованого групування полягає у об'єднанні окремих діапазонів коефіцієнтів подібності у групи за числовими значенням, що дозволяє здобути інформацію про розміри окремих груп та прослідкувати залежність між ознаками, що покладені в основу групування.

IV. РЕЗУЛЬТАТИ

В основу дослідження покладено два програмні продукти Unicheck та Plagiat.pl, які використовуються у Прикарпатському національному університеті для перевірки на плагіат студентських та наукових робіт та з виробниками яких укладено відповідні угоди Міністерством освіти і науки України (МОН).

Результатом роботи даних сервісів є формування підсумкового звіту (рис. 1):

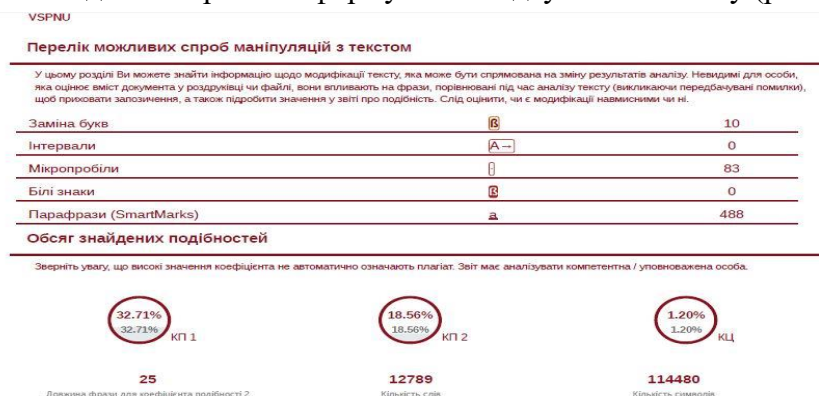


Рисунок 1 – Вікно результатів обробки тексту на плагіат

Основними елементами звіту служать два коефіцієнти подібності КП1 та КП2, які дають можливість визначити наступні типи запозичень:

- процентну частку аналізованого документу, що є ідентичною виявленим джерелам;
- список джерел фрагментів, позначених як скопійовані;
- повний текст перевіреного документу, з чітко визначеними виявленими запозиченнями;
- тривожний сигнал, що інформує про викривлення в тексті, в результаті використання різних алфавітів, що може вказувати на спробу приховання плагіату.

Беручи до уваги звіти, сформовані за результатами перевірки кваліфікаційних робіт у програмних сервісах Unicheck та Plagiat.pl було виконано аналіз дипломних та магістерських робіт протягом двох навчальних років (2020-2021 та 2021-2022 н.р.) у Прикарпатському національному університеті. В результаті проведеного дослідження були отримані статистичні дані, які відображають динаміку зміни якості навчальних матеріалів та відсоток текстових запозичень (рис. 2):

Динаміка						
Підрозділ	2020-2021		2021-2022		Прогрес	
	% якості	Середній %	% якості	Середній %	% якості	Середній %
Факультет математики та інформатики	74,51	19,25	91	10,7	16,49	-8,55
Факультет історії, політології і міжнародних відносин	80,65	17,66	84,2	12,6	3,55	-5,06
Фізико-технічний факультет	81,08	15,3	83,8	13,6	2,72	-1,7
Факультет фізичного виховання і спорту	90,62	17,29	93,3	13,6	2,68	-3,69
Факультет природничих наук	49,07	34,66	84,7	14,9	35,63	-19,76
Навчально-науковий Інститут мистецтв	58,97	25,01	80,4	15,2	21,43	-9,81
Факультет філології	75,86	20,05	80	18,4	4,14	-1,65
Економічний факультет	71,62	23,2	75,6	19,3	3,98	-3,9
Факультет психології	100	0	76,7	19,8	-23,3	19,8
Факультет туризму	37,5	38,6	72,9	20,5	35,4	-18,1
Педагогічний факультет	63,46	22,86	69,9	25,2	6,44	2,34
Інститут післядипломної освіти та довузівської підготовки	67,74	25,75	57,4	31,4	-10,34	5,65

Рисунок 2 – Динаміка зміни якості кваліфікаційних робіт у програмних сервісах Unicheck та Plagiat.pl

На основі застосування методу статистичного групування отримано динаміку зміни діапазону коефіцієнтів подібності за два роки (рис.3):

Динаміка (літні сесії, ОР бакалавр)			
Діапазон коефіцієнтів подібності		2020-2021	2021-2022
1	0% - 20%	50,6%	61,0%
2	20% - 40%	24,4%	19,4%
3	40% - 60%	13,0%	9,4%
4	60% - 100%	11,9%	10,2%

Рисунок 3 – Динаміка зміни діапазону коефіцієнтів подібності

Отримані статистичні обрахунки дозволили з однієї сторони зробити висновок про ефективність та подальшу доцільність застосування сервісів Unicheck та Plagiat.pl для перевірки кваліфікаційних робіт. З другої сторони на основі застосування даного методу

було виявлені взаємозв'язки і залежності між отриманими статистичними обрахунками та визначені типові способи використання плагіату студентами.

Зокрема в результаті проведеного детального аналізу досліджуваних робіт на факультеті математики та інформатики (опрацьовано звіти подібності чотирьох академічних груп (74 студенти)) було виявлено наступні, найбільш типові способи використання рерайтингу та копіпасти студентами:

- дослівне копіювання (зі списуванням включно);
- синонімічні заміни (синонімайзинг) із термінологічними замінами включно;
- зміна порядку слів;
- переклад з іншої мови;
- зміна ключових слів і фраз при збереженні основного змісту;
- введення сполучних слів і словосполучень, перехідних фраз, невеликих коментарів;
- плагіат даних (результати експериментів, емпіричних досліджень, розрахунків);
- плагіат зображень (фото, діаграми, схеми, креслення та ін.);
- запозичення програмних кодів.

З метою виявлення вказаних вище запозичень та унеможливлення подальшого використання вважаємо за необхідне використання вдосконалених методів та алгоритмів створення антиплагіатних програм за рахунок:

- перед обчисленням хеш-суми у методі шинглів виконувати сортування слів у шинглі, що дозволить покращити пошук запозичень у випадках, коли слова у тексті проаналізованої роботи були переставлені місцями;

- для збільшення ефективності використовувати модифіковані версії алгоритму шинглів, а саме:

- алгоритм MinHash [8], який кожній хеш-функції ставить у відповідність число – найменше значення для всіх шинглів досліджуваного рядка. В результаті для обробки отримуємо одновимірний масив (вектор) хеш-кодів довжин (короткі числові образи документу), завдяки чому значно зростає швидкість пошуку ймовірних дублікатів;
- алгоритм SimHash [9] – текст розбивається і кожен фрагмент, представлений у вигляді бінарного вектора хешується вибраною функцією. За рахунок того, що замість перевірки всього набору виконується порівняння окремих сигнатур, значно зростає швидкість алгоритму;

- щоб визначити заміну літер з кирилиці на латиницю на етапі нормалізації тексту застосовувати так званий метод синонімізації слів [3], який передбачає використання спеціальних словників (хоча проблемою використання цього методу залишається на сьогодні досить низька якість словників з української мови, які знаходяться у відкритому доступі, а якість комерційних словників важко перевірити через обмежений до них доступ);

- використовувати перехресну перевірку з допуском збігів, згідно з якою у «шингли» поміщати не хешсуми, а безпосередньо самі ланцюжки слів; а для більш детальної перевірки додатково використовувати словники синонімів та антонімів;

- коли потрібна більша точність перевірки застосовувати локальні алгоритми аналізу текстів як додаткові до методу шинглів (наприклад, Moodle Crot);

- для покращення роботи усіх методів на етапі нормалізації потрібно видаляти відомі словосполучення, враховувати синоніми, приводити літери до одного регістру та алфавіту;

- для пошуку міжмовних текстових збігів використовувати сучасні підходи з використанням моделей-трансформерів та граф-трансформерів;

- для запобігання копіювання чужих програмних кодів рекомендувати перевіряти програмні продукти з використання інструменту Copyleaks Programming Code Checker, який на основі штучного інтелекту може знаходити схожі шаблони коду в Інтернеті та базах даних.

V. ОБГОВОРЕННЯ

Як показує проведений аналіз сьогодні для перевірки наукових та навчальних праць наявність академічного плагіату доступно цілий ряд сервісів. Використання таких систем є вимогою часу та МОН. Проте дуже важливо розуміти, що не існує програм, які забезпечують 100-відсоткову перевірку тексту на плагіат. Антиплагіатні рішення не виявляють академічного плагіату, а лише допомагають його шукати.

VI. ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

Подальшим напрямком дослідження буде розробка програмного сервісу на основі застосування нових алгоритмів автоматизованого аналізу, який дозволить більш якісно проводити перевірку студентських робіт та наукових праць викладачів для виявлення академічного плагіату.

VII. ВИСНОВКИ

В роботі виконано аналіз відомих методів та алгоритмів, які використовуються для програмного пошуку текстових запозичень.

На основі проведеного статистичного дослідження щодо аналізу студентських робіт на предмет виявлення академічного плагіату визначено динаміку зміни якості кваліфікаційних робіт та виділено недоліки в існуючих антиплагіатних програмах, що дозволило стверджувати про доцільність подальших розробок сучасних програмних рішень та проведення еволюційно-проектних вдосконалень в області дотримання академічної доброчесності.

ЛІТЕРАТУРА

- [1] Верховна Рада України. Закон України про вищу освіту: документ 1556-VII (28 груд. 2014, зі змінами 25 верес. 2020). [Електронний ресурс]. Режим доступу: <https://zakon.rada.gov.ua/laws/show/2145-19#Text>. Дата звернення: Жовт. 26, 2022.
- [2] О.А. Куценко, О.А. Сердюк, "Огляд алгоритмів пошуку плагіату у програмному коді," Вісник Черкаського національного університету імені Богдана Хмельницького, Вип. 1, С. 75 – 85, 2019. DOI 10.31651/2076-5886-2019-1-75-85.
- [3] А.О. Білощицький, О.В. Діхтяренко, "Ефективність методів пошуку збігів у текстах," Управління розвитком складних систем, № 14, С. 144–147, 2013.
- [4] A. Kolcz, A. Chowdhury, J. Alspector, "Improved Robustness of Signature-Based Near-Replica Detection via Lexi-con Randomization, " *KDD 2004*, 22-25 August, 2004, Seattle, Washington, USA.
- [5] A. Chowdhury, O. Frieder, D. Grossman, M. McCabe, "Collection statistics for fast duplicate document detection, " *ACM Transactions on Information Systems (TOIS)*, vol. 20, Issue 2, April 2002.
- [6] The Illustrated Transformer. [Електронний ресурс]. Режим доступу: <https://jalammar.github.io/illustrated-transformer/>. Дата звернення: Лист. 08, 2022.
- [7] А.Т. Мармоза. Теорія статистики: Підручник. 2-ге вид. перер. та доп. К.: Центр учбової літератури, 2013, 592 с.
- [8] A. Shrivastava, "Simple and efficient weighted minwise hashing, " *Advances in Neural Information Processing Systems*, pp. 1498–1506, 2016.
- [9] Moses S. Charikar, "Similarity estimation techniques from rounding algorithms," *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, 2002, p. 380.