УДК 004.03

Л. М. Куперштейн
М. Я. Мельник

АНАЛІЗ МЕТОДІВ ПЕРЕВІРКИ ТЕКСТУ НА ПЛАГІАТ

Вінницький національний технічний університет

Анотація

Розглянуто методи перевірки текстів на плагіат та способи обману систем анти-плагіату. Наведено загальну характеристику етапів роботи системи перевірки на плагіат. Проаналізовано основні методи і визначено їх переваги та недоліки.

Ключові слова: плагіат, аналіз текстів, виявлення плагіату.

Abstract

Methods of checking texts on plagiarism and methods of deception of anti-plagiarism systems are considered. The general description of the stages of the system of checking for plagiarism is presented. The basic methods are analyzed and their advantages and disadvantages are determined.

Keywords: anti-plagiarism, text analysis, plagiarism detection.

Вступ

Проблема академічного плагіату на сьогодні дуже актуальна. З розвитком сучасних технологій та Інтернету рівень списування зростає великими темпами. Майже на будь-яку тему можна знайти готову роботу на спеціальних сайтах-архівах та скористатися її змістом, приховавши факт запозичення через заміну літер одного алфавіту на інший, додавання або видалення малозначимих слів, перестановки слів у реченнях та зміною роду і чисел на слова, а також навпаки.

З метою боротьби з проблемою було розроблено багато систем перевірки текстів на плагіат для виявлення у них запозичень, як безкоштовних, так і платних [1]. Більшість з них використовують власні алгоритми аналізу текстів та враховують методи приховування плагіату.

Метою роботи є аналіз методів та засобів перевірки текстових документів для більш ефективного використання та прийняття рішення щодо їх подальшого вдосконалення.

Результати дослідження

Плагіат – видача чужого наукового тексту за власний або використання такого тексту в своїх роботах, не вказуючи посилання на автора. Також розрізняють термін запозичення — використання у тексті роботи фрагментів другої роботи [2]. У Законі України "Про авторське право та суміжні права" плагіат визначається як оприлюднення (опублікування), повністю або частково, чужого твору під іменем особи, яка не є автором цього твору [3].

Алгоритми пошуку плагіату у текстах можна розділити на дві основні категорії: глобальні та локальні [4]. Глобальними називають ті, які використовують певні знання про усі документи, що розглядаються, а інші — локальні. Основна ідея локальних алгоритмів зводиться до синтаксичного аналізу документа. Наприклад, можна обчислити хеш-функцію від конкатенації двох найдовших речень, знайдених у тексті [4]. Перевагами такого підходу є його швидкодія та те, що можна зробити висновок про наявність плагіату у документі. До недоліків можна віднести його негнучкість, а також відсутність можливості перевірки усіх речень в тексті при використанні оригінальної варіації алгоритму.

Більш ефективним можна вважати метод частоти слова (term frequency). Обраховується відношення числа входжень певного слова до усієї кількості слів у тексті. Недостатньо точний, але дозволяє виявити плагіат у випадках, коли текст скопійованої роботи не змінювався кардинально.

У системах перевірки плагіату використовується також алгоритм Moodle Crot [5]. Його суть полягяє у видаленні з тексту документа усіх слів, що мають довжину менше трьох символів, небуквені знаки, пробіли, дефіси, крапки, коми тощо. Таким чином, отримуємо ланцюжок букв. Потім він з кроком n ділиться на частини по N символів у кожній. Далі потрібно обчислити хеш-

функцію від кожної частини і результат обчислення зберігається в набір. Далі порівнюються два різних набори хеш-сум. Найбільша точність алгоритму буде при n=1, але при цьому він буде працювати найменш продуктивно.

Перевагою даного алгоритму можна назвати точність та гнучкість. Недоліком є те, що при збільшенні точності до n=1, його продуктивність значно падає. Тому Moodle Crot не доцільно використовувати при швидкій перевірці текстів.

Метод шинглів найпопулярніший та найшвидший при пошуку плагіату в довільних текстах [5]. Саме його варіації використовують пошукові машини та сервіси аналізу текстів на плагіат.

Шингли − послідовності слів. Спочатку з тексту видаляються усі сполучники, спеціальні символи, короткі слова, крім пробілів. Далі з отриманих слів складаються самі шингли — послідовності. Плагіатом можна вважати збіг у 6 слів, тому коротші послідовності не враховуються. Від кожного шингла обраховується хеш-сума, потім створюється набір хеш-сум документа. Аналогічна операція виконується для другого документа. Таким чином, порівнюються шингли, і якщо у документах співпадають один або більше шинглів, тексти можуть вважатися подібними.

Існує ряд модифікацій алгоритма на основі шинглів [4]. Наприклад, одна із модифікацій базується на сортуванні, тобто перед обчисленням хеш-суми слова у шинглі будуть відсортовані. Це покращує пошук запозичень у випадках, коли слова у тексті проаналізованої роботи були переставлені місцями для обману системи анти-плагіату.

До переваг простого алгоритму шинглів можна віднести гнучкість, точність, швидкість та популярність. Проте недоліком є те, що в оригінальній версії порівнюється довільна кількість хеш-сум шинглів, а не всіх, тому нема можливості вказати всі конкретні місця плагіату в документі. На рис. 1 представлені етапи роботи системи анти-плагіату, яка використовує алгоритм шинглів.

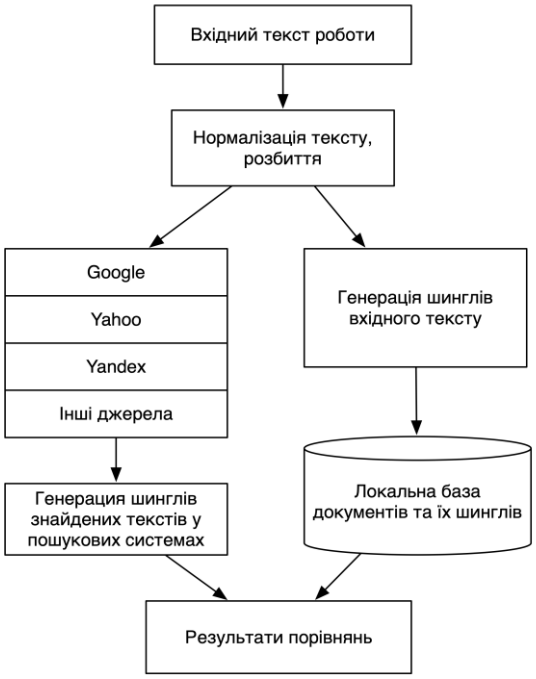

Рис. 1. Загальна схема алгоритму шинглів у системі виявлення плагіату

Етап нормалізації тексту призначений для виявлення та видалення прийомів обману систем-анти плагіату. Прийоми можна розділити на 2 основні категорії: обман людини та обман системи [5].

Для приховування факту плагіату від людини використовують перестановки слів, абзаців, додавання слів, що не несуть змісту, але візуально дозволяють тексту стати унікальним. Знаючи про можливу перевірку робіт автоматизованими алгоритмами, плагіатори використовують прийоми, спрямовані саме на обман програми.

Найпростіший варіант — заміна літер з кирилиці на латиницю. Цей прийом досить очевидний, як і перестановка слів. Тому на етапі нормалізації тексту (рис. 1), виконуються перетворення та сортування.

Відомий також метод синонімізації слів [6]. Але програми-синонімайзери майже завжди не враховують контекст вживання слова, тому початковий зміст втрачається. Важливий недолік даного способу − мала кількість українських словників у загальному доступі.

Найбільш реальний спосіб приховати плагіат на сьогодні – рерайт [6]. Тобто переписування оригінальної роботи так, щоб структурно вона відрізнялась, але зміст був той самий. Якість рерайту може бути низькою, наприклад, проста перестановка слів, заміна прямої мови на непряму, використання синонімів.

У випадку якісного рерайту, зміст передається той ж самий, під іншим кутом зору, але маніпулює тими ж фактами, що і в джерелі. Саме такий рерайт іноді неможливо визнати плагіатом, тому що потрібно довести, що при різних на вигляд текстах не було додано нових досліджень і фактів з теми. Рерайт вимагає багато часу та праці людини — це основна відмінність від наведених вище прийомів приховування плагіату у текстах. Реалізувати його автоматичним шляхом наразі неможливо.

Після виконання етапу нормалізації та розбиття тексту на речення (рис. 1), залежно від налаштувать система антиплагіату шукає тексти за реченнями у пошукових системах і паралельно генерує їх шингли для порівняння з локальною базою. Отримані від пошукових систем короткі відповіді (сніпети) також проходять процес генерації шинглів. Потім шингли документа, що перевіряється, порівнюються із шинглами знайдених в Інтернеті текстів і в локальній базі шинглів документів. Якщо шингли співпадають, робиться висновок про наявність плагіату та розраховується його відсоток.

Висновки

Визначити плагіат, якщо він був якісно перероблений (рерайт) — практично неможливо автоматизованими засобами. У якості алгоритму швидкої перевірки текстів на плагіат найбільш ефективним виявився метод шинглів із сортуванням. Алгоритм Moodle Crot доцільно використовувати, коли потрібна більша точність перевірки та не критична швидкодія. Доцільно застосовувати локальні алгоритми аналізу текстів як додаткові до методу шинглів та Moodle Crot. Вони можуть знайти плагіат, але не достатньо гнучкі.

Для ефективнішого аналізу текстів за допомогою алгоритму шинглів потрібно використовувати його покращену версію із сортуванням. Якщо звичайний алгоритм знаходить лише збіг і послідовність, то для покращеної версії послідовність не відіграє ролі.

Moodle Crot потребує вдосконалення щодо підбору оптимальних значень n та N. Адже існують багато словосполучень по 2-3 слова, які не можна вважати плагіатом. На етапі нормалізації тексту доречно видаляти такі словосполучення, використовуючи словники. Також можна задавати значення n та N залежно від тематики тексту, так як середня довжина слова у, наприклад, інтерв'ю менша ніж в наукових документах. Дані прийоми дозволяють покращити точність та ефективність алгоритму.

Загальна ефективність методів аналізу текстів залежить від того, у якому саме вигляді був поданий до них текст. Для покращення роботи усіх методів на етапі нормалізації потрібно видаляти відомі словосполучення, враховувати синоніми, приводити літери до одного регістру та алфавіту.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Чиркин Е.С. Системы автоматизированной проверки на неправомерные заимствования // Вестник ТГУ. – 2013. − №12. − С. 164-171.
2. Петренко В.С. Поняття плагіату [Електронний ресурс]. — Режим доступу: http://www.clj.nuoua.od.ua/archive/14/29.pdf.
3. Закон України "Про авторське право і суміжні права" [Електронний ресурс]. — Режим доступу: https://zakon.rada.gov.ua/laws/show/3792-12.
4. Зеленков Ю.Г, Сегалович И.В. Сравнительный анализ методов определения нечетких дубликатов для Web-документов [Електронний ресурс]. – Режим доступу: http://rcdl2007.pereslavl.ru/papers/paper_65_v1.pdf
5. Білощицький А.О., Діхтяренко О.В. Ефективність методів пошуку збігів у текстах // Управління розвитком складних систем. − 2013. −№ 14. − С. 144-147.
6. Методы обхода антиплагиата [Назва з екрану]. — Режим доступу: http://antiplag.ru/blog/neskolko-effektivnyx-metodov-obxoda-proverki-na-antiplagiat.

Куперштейн Леонід Михайлович — к.т.н., доцент, кафедри захисту інформації, Вінницький національний технічний університет, м. Вінниця

Мельник Максим Ярославович — студент групи 1БС-15б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: aktotut@pm.me

Kupershtein Leonid M. — Ph.D, Assoc. professor, Information Protection Chair, Vinnytsia National Technical University, Vinnytsia

Melnik Maxim Y. — student 1BS-15b, Faculty of Information Technologies and Computer Engineering, Vinnystia National Techincal University, Vinnytsia, e-mail: aktotut@pm.me