УДК 518.14

ВИКОРИСТАННЯ МЕТОДІВ КОМП'ЮТЕРНОЇ ЛІНГВІСТИКИ ПРИ ВСТАНОВЛЕННІ АВТОРСТВА ТЕКСТУ

Ванін В.В., д.т.н.,
vaninvladimir30@gmail.com, ORCID: 0000-0001-7008-7269
Залевська О.В., к.т.н.,
o.zalevska@kpi.ua, ORCID: 0000-0002-3163-1695
Савчук Б.І.
antipich69@gmail.com, ORCID: 0000-0002-5399-3267
Ситник А.
Sytnik.akim@gmail.com, ORCID: 0000-0001-8085-2163
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського» (Україна)
Zhu Shiwei
zhusw@sdas.org, ORCID: 0000-0002-2875-0706
Information Research Institute, Qilu University of Technology (Shandong Academy of Sciences), (Jinan, China)

На цей час не існує ні сервісів, ні окремих програм, алгоритмів та модулів, робота яких полягала у співставленні текстів власника на встановлення того, чи є власник автором усіх текстів, які перевіряються. Надалі буде йти мова саме про потенційний алгоритм, ідею роботи якого викладено вище. Перевірки n-ої кількості текстів на те, чи автор у них один або декілька, є задачею для якої не обов'язково використовувати потужне устаткування та витрачати час та ресурси на навчання окремої нейронної мережі, завданням якої було б намагатися віднайти схожі об'єкти у величезній базі даних.

*У роботі описано метод швидкого аналізу тексту на предмет встановлення авторства шляхом поділу тексту на елементи та їх окремий аналіз без використання великої кількості часу на обробку даних. Розроблено програму для розрахунків та візуалізації проведеного аналізу тексту, який може внесений в програми в розповсюдженому форматі *.doc або *.docx. Досліджено результати такого аналізу на низці робіт різних авторів та різної тематики робіт. Роботи можуть бути порівняні в рамках одного автора на предмет цілісності авторських робіт або декількох авторів між собою на предмет можливого запозичення матеріалу. Алгоритм не дає гарантованої відповіді та може бути використаний лише як підстава для додаткової перевірки робіт.*

На даний момент є актуальним перевірка великої кількості тексти з причини потреби встановлення його оригінальності. Є типовим вважати, що за випадком, коли текст є оригінальним, а саме його оригінальність 90% і більше, матеріал є працею автора. З іншого боку,

потрібно зазначити, що перевірка матеріалу на оригінальність дуже ресурсозатратна.

Запропонований алгоритм є актуальним, оскільки здатний розширити набір можливостей існуючих сервісів для встановлення оригінальності текстів, а також зменшити навантаження на їх обчислювальну потужність, оскільки більшість вже розроблених варіантів використовую алгоритми штучного інтелекту, швидкість якого залежить як від алгоритму реалізації, так і від потужностей головної системи.

Ключові слова: аналіз тексту, антиплагіат, достовірність авторства, комп'ютерна лінгвістика

Постановка проблеми. Процес обробки вимагає пошуку відповідностей по великих за обсягом баз даних інформації, а також обчислювальна обробка для розділення окремих частин тексту на окремі фрази. Для таких потреб використовують потужні обчислювальні ресурси, а також розробляють складі алгоритми, які включають в себе використання нейронних мереж. Останні також потребуються як і додаткове місце в пам'яті для збереження проміжної інформації а також і деяку обчислювальну потужність. Важливим є зазначити, що кожний екземпляр матеріалу, що оброблюється, має бути скопійованим ресурсу загальної інформаційної системи для розширення її бази інформаційних даних.

Все це приводить до обмеженої швидкості обробки та перевірки матеріалу багатьох авторів на оригінальність та не свідчить про відповідне авторство того, хто ініціалізує перевірку файлу. Отже, існує потреба в розробці такого алгоритму, результатом роботи якого було б встановлення авторства власника тексту.

Аналіз останніх досліджень і публікацій. Ідея алгоритму полягає в пошуку ключових слів, які надалі будуть називатися «ключами». Такі «ключі» є словами основної частини мови, а також вказівками на те, як часто та з яким наповненням їх використовує автор [1].

Сформований набір «ключів» надалі має порівнюватися із подібним набором «ключів», яких є результатом аналізу іншого тексту вірогідного автора [2].

Окремо зберігаються «ключі» іменних частин мов для того, щоб у разі потреби можна було перевірити достовірність використаної термінології, а саме викласти припущення, чи є текст псевдонауковим. Достовірність термінології перевіряється за рахунок пошуку ключових слів у визначенні слова, яке перевіряють. Ключовими словами є тема тексту, а також інші ключі, які перевіряються на достовірність використаної термінології [3-4].

Час на перевірку матеріалу є значно нижчим, що робить алгоритм більш доступним та легким у використанні, бо перевірка матеріалу може бути розділена за типом пошуку [5]:

1) встановити автора;
2) перевірити авторство.

Перший варіант роботи алгоритму є доцільно використовувати, коли, наприклад, потрібно встановити відповідність того, що окремий розділ виконав деякий автор, а не інший, наприклад, автор іншого розділу. Другий варіант – типовий варіант встановлення відповідності авторства матеріалу на основі n-ої кількості додаткових екземплярів, що є підтвердженими на авторство особи, матеріал якої перевіряє алгоритм.

Недоліком для такого алгоритму є можливий порушений синтаксис у тексті вхідних робіт або його навмисне ігнорування, наприклад, використання таблиць, де дотримання синтаксису між записами не є необхідним.

Формування цілей статті. Розробка алгоритму обробки тексту для встановлення його автора.

Основна частина. Перевагою запропонованого алгоритму є кількість інформації, необхідної для збереження, а також можливість встановлення авторства навіть невеликої кількості тексту розміром з абзац або іншими словами: 5-10 речень, як на приклад.

Відповідні «ключі», що є важливим зазначити, зберігаються у вигляді бази даних у вигляді таблиць, розділених по частинам мови та додатковим «ключам», а також таблиці накопичених авторів із встановленням зв’язку автора до теми його текстів, а також їх «ключів». База даних може зберігатися як локально, так і на віддаленому сервері. У випадку користування віддаленим сервером можлива відсутність вказання авторства для наборів «ключів», власна база даних може бути сформована із уже існуючих баз даних текстів, що зберігаються на сервісах, або з їх джерел.

Передбачено використання файлів формату *.doc та *.docx. Після імпорту файлу або файлів, вони конвертуються до текстового формату та записуються у тимчасову зміну, після чого клас KeysBuilder в своєму конструкторі поступово заповнює об’єкт класу Keys на основі методів _cut_sentences, _cut_words, _cut_newlines. Після оброки вхідної інформації дані ключів можуть бути збережені як до бази даних на прикладі PostgreSQL або MySQL так і експортовані у вигляді файлу формату *.json, з подальшою можливістю імпорту без повторної обробки.

Результатом роботи вище описаного алгоритму є список значень, які можна використати для абсолютного відношення або в порівняння в моделями інших значень, отриманих з інших робіт для установлення авторства особи окремих частин тексту, розділених на абзаци, а також перелік зауважень щодо використаної термінології, стилізації тексту а також відношення кількості зайвого наповнення тексту до його розміру.

Окрім цього також надаються графіки використання літер та порівняння групи середніх значень (рис. 1) або дані по ключам для одного об’єкта (рис. 2).

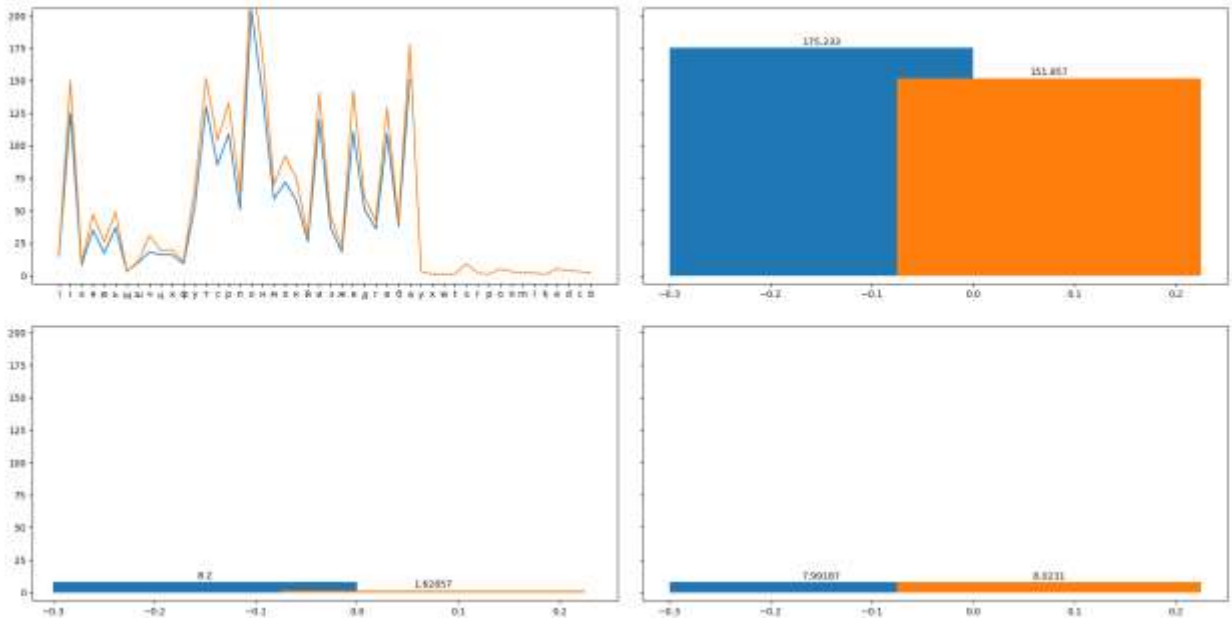

Рис. 1. Зліва направо: перший ряд – порівняння використання літер, порівняння середньої довжини речення; другий ряд – порівняння середньої кількості слів у реченні та середньої довжини слова

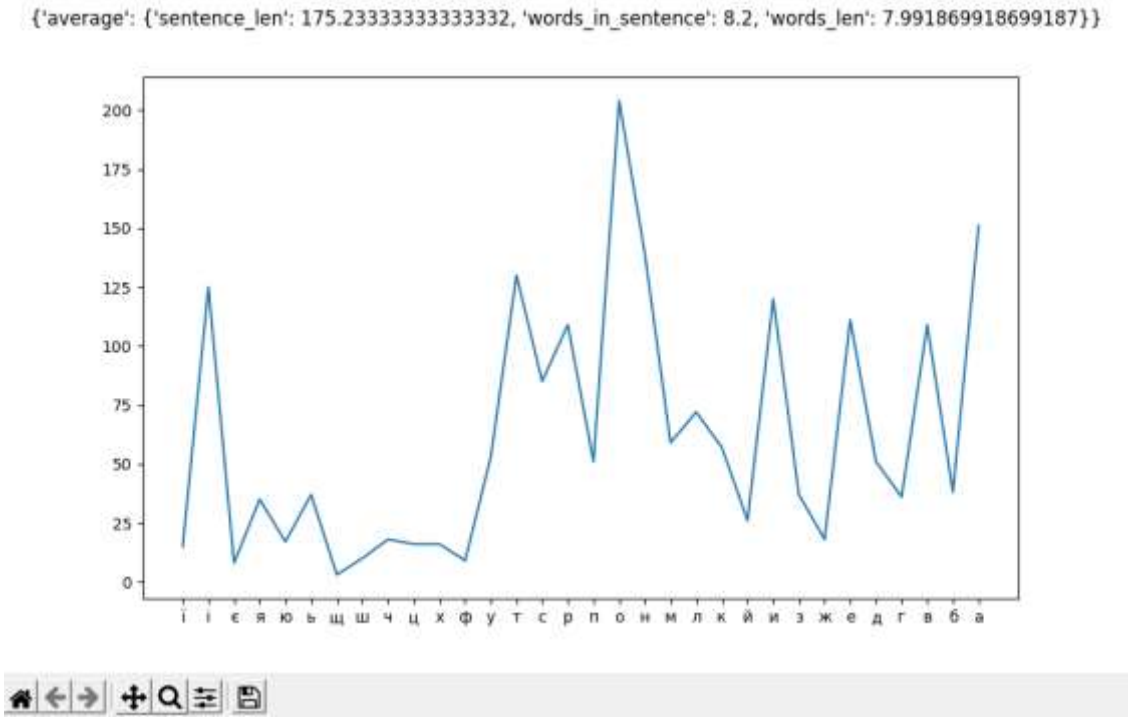

Рис. 2. Дані по ключам одного об’єкта

Для прикладу було порівняно тексти з різним наповненням, але присутніми повторними елементами.

Отриманий результат не є приводом не вважати особу автором тексту, але є причиною потреби додаткової перевірки на оригінальність та встановлення можливого авторства іншої особи.

Було порівняно матеріали різного складу та одного автора на

предмет запозичення матеріалу зі стороннього ресурсу рис. (3-6).

Синім кольором – перша (1) робота, оранжевим – друга (2) робота: того ж автора, але з іншим контекстом, зеленим – третя (3) робота, яка була основою для написання роботи другої.

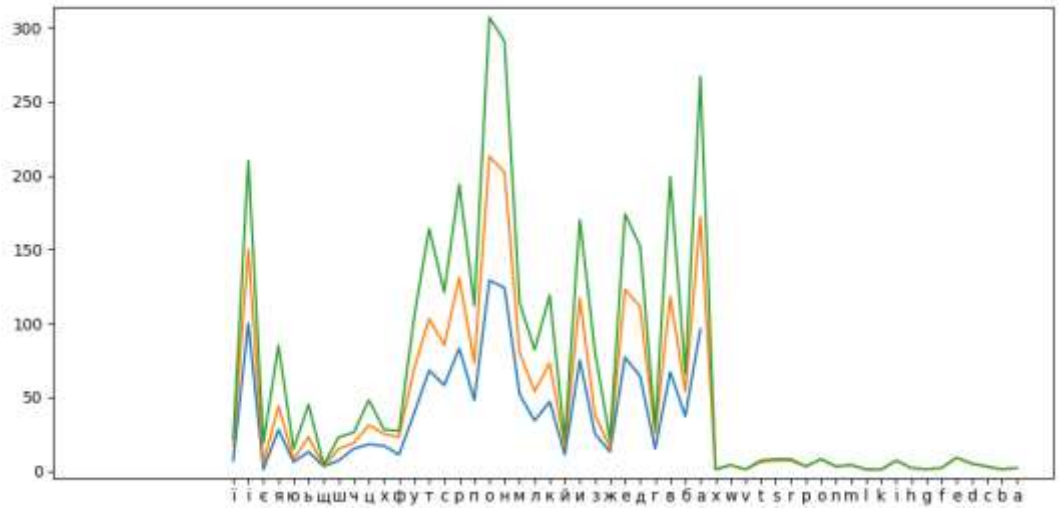

Рис. 3. Порівняння використання літер

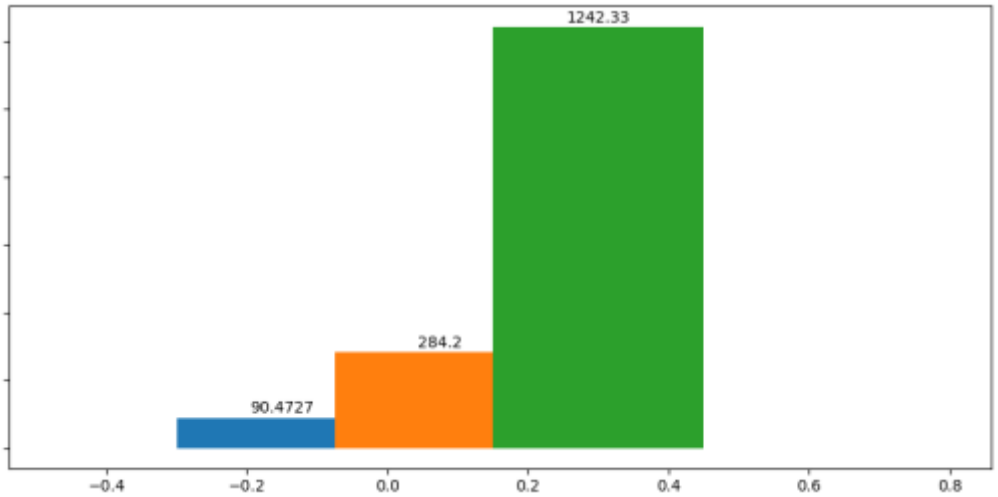

Рис. 4. Порівняння середньої довжини речення

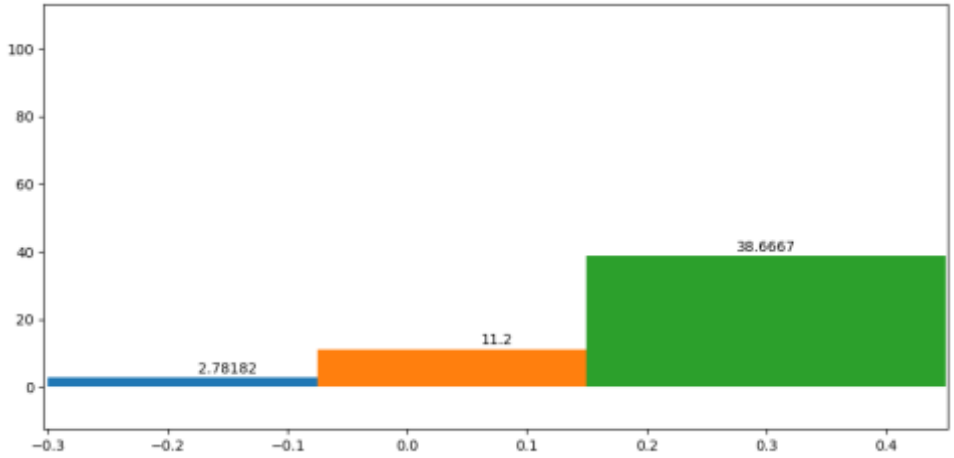

Рис. 5. Порівняння середньої кількості слів у реченні

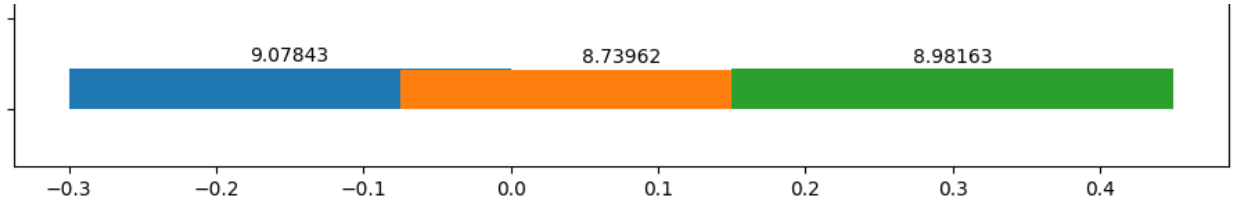

Рис. 6. Порівняння середньої довжини слова

Із результатів аналізу трьох робіт можна зробити висновок, що хоч і робота (1) схожа на роботу (2), але в роботі (2) є можливим використання елементів із роботи (3). Також роботи мають схожу тематику, тобто лексичне наповнення. Таке припущення можна зробити, виходячи з результатів на Рисунку 6.

Висновки. В результаті дослідження роботи алгоритму можна зробити висновок про достовірність його роботи та правильність вибраного підходу до аналізу тексту на основі порівнянь його кількісних характеристик, що може збільшити швидкість обробки великої кількості даних.

Література

1. Matplotlib - User Guide. Режим доступу: https://matplotlib.org/stable/users/index.html.
2. Shah A. Python-docx2txt. Режим доступу: https://github.com/ankushshah89/python-docx2txt.
3. Herdan Gustav. The advanced theory of language as choise and chance. *Kommunikation und Kybernetik in Einzeldarstellungen*. 1966. pp. 14–437.
4. Bobadilla J., Ortega F., Recommender systems survey. *Knowledge- Based Systems*. 2013. pp. 109-132.
5. Stamatatos E. Author identification: Using text sampling to handle the class imbalance problem. *Information Processing and Management*: an International Journal. 2008. No 44. pp.790–799. Режим доступу: https://dl.acm.org/citation.cfm?id=1347518.

THE USE OF COMPUTER LINGUISTICS METHODS IN ESTABLISHING THE AUTHORSHIP OF THE TEXT

Volodymyr Vanin, Olga Zalevska, Bohdan Savchuk, Akym Sytnyk, Zhu Shiwei

Currently, there are neither services nor individual programs, algorithms and modules whose job is to compare the owner's texts to establish whether the owner is the author of all the texts being checked. In the future, we will talk

about a potential algorithm, the idea of which is described above. Checking the n-th number of texts to determine whether they have one or several authors is a task for which it is not necessary to use powerful equipment and spend time and resources on training a separate neural network, whose task would be to try to find similar objects in a huge database.

*The paper describes a method of fast analysis of text to establish authorship by dividing the text into elements and their separate analysis without using a large amount of time for data processing. A program for calculating and visualizing the analysis of the text has been developed, which can be entered into programs in the common format *.doc or *.docx. The results of such analysis were investigated on a number of works by different authors and different subjects. Works can be compared within the framework of one author for the integrity of the author's works or several authors among themselves for possible borrowing of material. The algorithm does not give a guaranteed answer and can be used only as a basis for additional verification of works.*

At the moment, it is relevant to check a large number of texts because of the need to establish its originality. It is typical to consider that in the case when the text is original, namely its originality is 90% or more, the material is the work of the author. On the other hand, it should be noted that checking the material for originality is very resource-intensive.

The proposed algorithm is relevant because it is able to expand the set of capabilities of existing services to establish the originality of texts, as well as reduce the load on their computing power, since most of the already developed options use artificial intelligence algorithms, the speed of which depends on both the implementation algorithm and the capacity of the host system.

Keywords: text analysis, anti-plagiarism, authenticity of authorship, computer linguistics

References

1. Matplotlib - User Guide. URL: https://matplotlib.org/stable/users/index.html.
2. Shah A. Python-docx2txt. URL: https://github.com/ankushshah89/python-docx2txt.
3. Herdan Gustav. The advanced theory of language as choise and chance. *Kommunikation und Kybernetik in Einzeldarstellungen*. 1966. pp. 14–437.
4. Bobadilla J., Ortega F., Recommender systems survey. *Knowledge- Based Systems*. 2013. pp. 109-132.
5. Stamatatos E. Author identification: Using text sampling to handle the class imbalance problem. *Information Processing and Management*: an International Journal. 2008. No 44. pp.790–799. URL: https://dl.acm.org/citation.cfm?id=1347518.