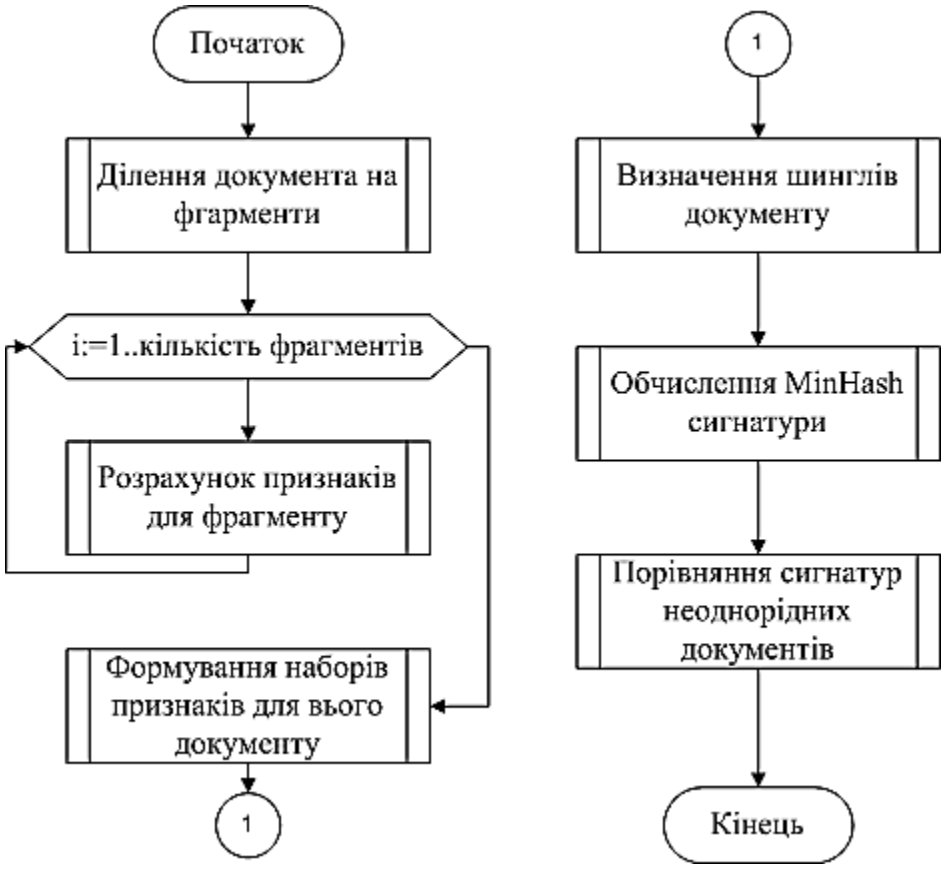

Рис. 1. Блок-схема системи виявлення плагіату

Список використаних джерел

1. Дикань С.А. Плагіат в освіті: походження, причини та шляхи подолання [Текст] / Дикань С. А., Безпека життєдіяльності. 2007. – №5. – Ст. 16-20.

2. Брін С., Девіс Д., Гарсія-Моліна Х. Механізми виявлення копіювання для електронних документів [Текст] / Vine. – 2001.

3. Хейнз Н., Масштабуємі відбитки документі [Текст] / USENIX Семіран по електонній торгівлі, листопад. 1996.

4. Васильвицький С. Робота з Великими Даними (записи з лекцій, університет Колумбії) [Текст] / 2011.

Різник Р.К.

студент,

Харківський національний університет радіоелектроніки

ВИЗНАЧЕННЯ ДОПОМІЖНОГО СЛОВНИКА ДЛЯ АНАЛІЗУ НЕЧІТКИХ ДУБЛІКАТІВ АЛГОРИТМОМ I-MATCH

В останні роки великі динамічні сховища документів стали звичайним явищем, чому сприяє швидкий розвиток інтернету. В силу багатьох факторів, таких як поширення спаму, плагіату та інших постає задача виявлення в

сховищах документів-дублікатів, не обов'язково абсолютно ідентичних, можливо з деякими змінами, але таких, які б розцінювалися як нечіткі дублікати.

На даний момент запропоновано безліч способів виявлення нечітких дублікатів. Їхні цілі варіюються від підвищення якості визначення до зменшення вимог до обчислювальних ресурсів та ресурсів пам'яті, необхідних для їх роботи. З масивними сховищами даних, швидкість обробки документів стає критичною, що робить техніки, які оперують одиничними хеш-сигнатурами документів, досить привабливими. Однією з таких технік є I-Match метод. Однак сигнатури, створені за допомогою цього методу досить чутливі до невеликих змін в тексті документів.

В даній роботі пропонується підхід, направлений на зменшення чутливості алгоритму I-Match до невеликих змін в документі, що базується на введенні допоміжного словника, для обчислення хеш-сигнатури.

Алгоритм I-Match обчислює лише одну хеш-сигнатуру якою представляється документа, це гарантує, що будь-який документ буде відображений в один і лише один кластер. Хеш-сигнатура обчислюється на основі множини слів, що входять в документ і в спеціальний заздалегідь визначений словник L. Процес генерації сигнатури може бути представлений наступним чином:

– будується словник L на основі множини всіх слів корпусу документів;

– для кожного документу d, визначається множина з унікальних слів U що містяться в цьому документі;

– обчислюється I-Match хеш-сигнатура на базі перетину множини слів основного словника і слів з документу d: $S = (L \cap U)$.

Ефективність алгоритму I-Match залежить від правильності побудови словника L. Одна з ефективних стратегій побудови словника полягає у виборі з усієї множини, слів с середнім значенням коефіцієнту idf – оберненої частоти документу. Так як слова з низьким коефіцієнтом idf (часто вживані) та слова з високим коефіцієнтом idf (рідко вживані) зменшують якість роботи алгоритму при визначенні дублікатів, то вони виключаються зі словника L. Слід зазначити, що вибірка слів з високим коефіцієнтом idf може бути дуже ефективною при пошуку конкретного документу, але вона також може захопити слова з помилками, що знижує їх цінність при виявленні нечітких дублікатів. Не існує чітких рекомендацій щодо вибору точок відсікання слів з низьким та високим коефіцієнтом idf, вони вибираються методом проб і помилок.

I-Match може привести до помилкових спрацювань, якщо множина слів документу має досить малий перетин із словником L. Пропонується розширення алгоритму I-Match, що дозволить в деякій мірі вирішити проблему недостатньої потужності перетину. Словник L представляє впорядкований набір всіх слів за idf коефіцієнтами, з нього відкидаються слова з низькими коефіцієнтами idf, на основі яких будується новий словник B, слова в якому упорядковуються по зростанню коефіцієнтів. У модифікованій методиці I-Match, всякий раз, коли перетин слів документу з основним словником не задовольняє визначеному порогу, використовується вторинний словник. Оскільки вторинний словник B може бути набагато більшим за L і так як він

потенційно може містити багато нерелевантних слів, то слова в ньому впорядковуються в зворотному напрямку. Це гарантує, що слова з вторинного словника, що увійдуть до хеш-сигнатури документу будуть мати значення коефіцієнту idf максимально близькі до слів з основного словника. Словник L складається з множини слів, що знаходяться в наступному діапазоні: $[idf_{min}, idf_{max}]$. Слова, для яких коефіцієнт $idf < idf_{min}$ – це стоп-слова, які мають високу частоту вживання, та які ймовірно не пов'язані з суттю будь-якого конкретного документу, тому немає сенсу використовувати їх для обчислення хеш-сигнатури документа. Необхідно звернути увагу на слова, для яких коефіцієнт $idf > idf_{max}$. Ці слова зустрічаються рідше ніж ті, що включені до словника L, тому більша ймовірність обчислити точну хеш-сигнатуру документа.

Генерація хеш-сигнатури за допомогою модифікованого I-Match алгоритму включає наступні етапи:

– будується основний словника L, та вторинний словник B, на основі множини слів всього корпусу документів, що будуть використовуватися при обчисленні хеш-сигнатури документів. Словник L включає слова з діапазону $[idf_{min}, idf_{max}]$, словник B включає слова, для яких коефіцієнт $idf > idf_{max}$;

– для кожного документу d, визначається множина з унікальних слів U що містяться в цьому документі;

– обчислюється I-Match хеш-сигнатура, як перетин множини слів з основного словника і документу $S = (L \cap U)$;

– якщо $\frac{|S|}{|U|}$ нижче заданого порогу, словник S розширяється мінімальною кількістю слів із другорядного словника U щоб задовільнити обмеженню порога. Величина порогу визначається експериментально, та може відрізнятися для різних типів документі (наприклад різні для електронних листів, веб сторінок та ін.);

– якщо в словнику U не достатньо слів для подолання порогу $\frac{|S|}{|U|}$, генерується NULL сигнатура документа.

Необхідно зауважити, так як словник U може бути досить великим, необхідно визначити його розмір таким чином, щоб забезпечити баланс між використанням доступних обчислювальних ресурсів (оперативної пам'яті) та кількістю слів, щоб уникнути генерації NULL сигнатур.

Список використаних джерел

1. Bilenko M., Mooney RJ. (2002): Learning to combine trained distance metrics for duplicate detection in databases. Technical report AI 02-296.

2. Conrad J., Guo X., Schriber C. (2003): Online duplicate document detection: signature reliability in a dynamic retrieval environment.

3. Henzinger M. (2006): Finding Near-Duplicate Web Pages: a Large-Scale Evaluation of Algorithms.

4. H. Wu and R. Luk and K. Wong and K. Kwok. (2008) «Interpreting TF-IDF term weights as making relevance decisions». ACM Transactions on Information Systems, 26 (3).