ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА КОМП'ЮТЕРНА ТЕХНІКА

УДК 004.021

М. О. Гранік[1]
В. І. Месюра[1]

АЛГОРИТМ ВИЗНАЧЕННЯ СХОЖОСТІ ТЕКСТІВ НОВИН НА ОСНОВІ ПОЛІНОМІАЛЬНОГО ХЕШУВАННЯ

[1]Вінницький національний технічний університет

Запропоновано алгоритм порівняння текстів новин на основі поліноміального хешування. Алгоритм може бути використаний для кластеризації текстів новин.

Ключові слова: новини, порівняння новин, поліноміальне хешування.

Вступ

Проблема визначення схожості текстів новин є дуже актуальною. Отримання чисельної міри схожості текстів новин може бути ефективно використане для задачі кластеризації новин. Кластеризація новин, у свою чергу, є важливою практичною проблемою, адже результати її розв'язання можуть бути використані у агрегаторах новин та у системах оцінювання правдоподібності інформації текстів новин.

Визначення схожості текстів (не обов'язково текстів новин) також є важливою практичною проблемою. Таке визначення широко використовується у біоінформатиці (наприклад, у генній онтології). Переважно у цій області його використовують для визначення схожості генів, протеїнів тощо.

Визначення чисельної міри схожості текстів також використовується для пошуку інформації, класифікації текстів, автоматичного визначення теми текстів, автоматичної генерації запитань та відповідей на них, машинного перегляду, анотування тексту (визначення підмножини тексту, що передає його основну ідею) [1].

Існує декілька шляхів визначення схожості текстів. Серед них виділяють визначення косинусного коефіцієнта, обчислення коефіцієнта Жаккара, коефіцієнта Соренсена, коефіцієнта Сімпсона, коефіцієнта Браун-Бланке, коефіцієнта Кульчинського [1, 2].

Недоліком перерахованих методів є те, що вони краще працюють для порівняння множин, ніж саме для порівняння текстів. Якщо використовувати їх без жодних модифікацій, то слова, вжиті у різних відмінках, числах, тощо будуть вважатись різними елементами множин.

Метою роботи є розробка алгоритму, позбавленого цього недоліку.

Поліноміальне хешування

Хешування рядків — процес встановлення однозначної відповідності між рядком та чисельним значенням з певного фіксованого чисельного проміжку. Варто зазначити, що одному рядку завжди відповідає лише одне чисельне значення, однак тому самому чисельному значенню можуть відповідати декілька рядків.

У поліноміальному хешуванні значення хешу для заданого рядка довжиною n обчислюється за формулою

$$H = \left(\sum_{i=1}^{n} a_i \cdot p^i \right) \bmod, \qquad (1)$$

де a — коефіцієнти, числові значення яких поставлені у відповідність кожному символу рядка (наприклад, номер у таблиці ASCII); p — показник многочлена; M — число, що визначає інтервал, до якого належатиме хеш (очевидно, що можливі значення хешу належать числовому діапазону [0; M – 1]). Тобто рядку ставиться у відповідність поліном з коефіцієнтами, рівними числовим зна-

ченням кожного із символів рядка. Степінь полінома дорівнює довжині відповідного рядка.

Переваги поліноміального хешування є такими:

1) простота визначення;

2) можливість ефективного обрахунку поліноміального хешу заданого рядка (складність обчислення — $O(N)$, де N — довжина рядка);

3) можливість ефективного обчислення хешу будь-якого підрядка.

Поліноміальні хеші використовуються в багатьох алгоритмах, що працюють з рядками. Відомий приклад — алгоритм Рабіна-Карпа, мета якого — знайти всі позиції входження одного рядка в інший як підрядка.

Від вибору параметрів формули визначення поліноміального хешу залежить ймовірність колізії (тобто випадку, коли двом різним рядкам відповідають однакові значення хешу). Зазвичай, значення p обирають так, щоб це було просте число, більше за будь-яке значення коефіцієнтів a. Звичайно, збільшення значення M зменшує ймовірність колізії (однак збільшує обсяг пам'яті, необхідний для зберігання хешу рядка).

Якщо припустити, що значення полноміального хешу — рівномірно розподілена випадкова величина, то відповідно до парадоксу днів народження, ймовірність колізії для відносно невеликих значень M є достатньо великою. Так, для M = 1000000000 достатньо лише близько 30000 рядків для того, щоб колізія відбулась з ймовірністю 0,5, і приблизно 67000 рядків — для того, щоб вона відбулась з ймовірністю 0,9. Для уникнення колізій часто для одного рядка обчислюють декілька значень поліноміального хешу — наприклад, по двох різних модулях. Це зменшує ймовірність колізії. Інколи один або декілька з цих модулів вибирають випадковим чином.

Алгоритм визначення схожості текстів новин на основі хешування

Пропонується алгоритм визначення схожості текстів новин на основі хешування:

1. *Проведення операції стемінгу над усіма словами усіх текстів, що розглядаються.*

Стемінг — операція скорочення слів шляхом видалення із них неважливих частин, таких як префікс, суфікс чи закінчення (проте вважати, що в результаті застосування операції стемінгу кожне слово замінюється на його корінь, некоректно). Застосування алгоритмів стемінгу є поширеним у пошукових системах. Саме стемінг допомагає вирішити вищеописану проблему. Після його застосування різні словоформи вважатимуться одним і тим самим словом.

2. *Видалення стоп-слів.*

Стоп-слова (або шумові слова) — це такі слова у тексті, що не несуть змістовного навантаження. Під стоп словами зазвичай мають на увазі прийменники, частки, деякі інші окремі слова інших частин мови. Оскільки стоп-слова не несуть змістовного навантаження, їх врахування під час обчислення схожості текстів можуть суттєво спотворювати отримані результати.

3. *Прибирання з тексту усіх пробільних символів та пунктуаційних знаків.*

4. *Зведення отриманого тексту до нижнього регістру.*

5. *Розбиття тексту на синтаксичні n-грами.*

Синтаксичним n-грамом називається підрядок отриманого тексту з n символів. Мета цього етапу алгоритму — отримати вектор, кожний елемент якого є унікальним n-грамом (будемо вважати два n-грами різними, якщо відрізняються їх стартові позиції у тексті і не звертатимемо уваги на їхнє значення)

6. *Обчислення поліноміального хешу кожного із отриманих n-грамів.*

Таким чином, після цього кроку для кожного тексту буде отримано вектор чисел, кожне з яких є обчисленим значенням хешу для деякого n-граму із цього тексту.

7. *Визначення схожості текстів на основі коефіцієнта Жаккара для відповідних їм векторів, що зберігають хеші n-грамів.*

Коефіцієнт Жакара обчислюється таким чином. У відповідність кожному з векторів ставиться відповідна множина хешів. Коефіцієнт Жаккара визначається як частка від ділення потужності множини перетину двох отриманих множин на потужність множини об'єднання цих множин. Чим ближчим до одиниці є значення коефіцієнта Жаккара, тим більш схожими вважаються тексти. Цей метод є узагальненим методом порівняння двох множин на схожість. Саме отримане значення коефіцієнта і вважається мірою схожості текстів новин.

Розглянемо процес роботи алгоритму на прикладі двох уривків текстів новин. Цей приклад служить лише для прояснення деталей алгоритму, його числовий результат не може бути врахова-

ний під час оцінки коректності алгоритму з низки причин (по-перше, до уваги береться лише уривок тексту; по-друге, порівнюється лише одна пара текстів).

Перший текст взято з інформаційного агентства BBC:

England suffered their worst humiliation since they were knocked out of the 1950 World Cup by USA in Brazil as Iceland shocked them in the last 16 of Euro 2016

Другий текст взято із тексту новини агентства CNN:

Iceland pulled off one of the most astonishing results in the history of European football on Monday, knocking England out of the Euro 2016 finals.

Після виконання пунктів 1—4 алгоритму, отримуємо такі два тексти:

Текст агентства BBC:

englandsuffertheirworsthumilisinctheyknockout1950worldcupbyusainbrazilasicelandshocktheminlastof16euro2016

Текст агентства CNN:

icelandpulloffoneofmostastonishresultinhistoriofeuropeanfootbalonmondayknockenglandoutofeuro2016final

Розглядатимемо *n*-грами довжини 6. Тоді для першого тексту отримаємо таку множину *n*-грамів:

englan, ngland, glands, landsu, andsuf, ndsuff, dsuffe, suffer, uffert, fferth, ferthe, erthei, rtheir, theirw, heirwo, eirwor, irwors, rworst, worsth, orsthu, rsthum, sthumi, thumil, humili, umilis, milisi, ilisin, lisinc, isinct, sincth, incthe, ncthey, ctheyk, theykn, heykno, eyknoc, yknock, knocko, nockou, ockout, ckout1, kout19, out195, ut1950, t1950w, 1950wo, 950wor, 50worl, 0world, worldc, orldcu, rldcup, ldcupb, dcupby, cupbyu, upbyus, pbyusa, byusai, yusain, usainb, sainbr, ainbra, inbraz, nbrazi, brazil, razila, azilas, zilasi, ilasic, lasice, asicel, sicela, icelan, celand, elands, landsh, andsho, ndshoc, dshock, shockt, hockth, ockthe, ckthem, kthemi, themin, heminl, eminla, minlas, inlast, nlasto, lastof, astof1, stof16, tof16e, of16eu, f16eur, 16euro, 6euro2, euro20, uro201, ro2016

Для другого тексту ця множина *n*-грамів виглядає таким чином:

icelan, celand, elandp, landpu, andpul, ndpull, dpullo, pullof, ulloff, lloffo, loffon, offone, ffoneo, foneof, oneofm, neofmo, eofmos, ofmost, fmosta, mostas, ostast, stasto, taston, astoni, stonis, tonish, onishr, nishre, ishres, shresu, hresul, result, esulti, sultin, ultinh, ltinhi, tinhis, inhist, nhisto, histor, istori, storio, toriof, oriofe, riofeu, iofeur, ofeuro, feurop, europe, uropea, ropean, opeanf, peanfo, eanfoo, anfoot, nfootb, footba, ootbal, otbalo, tbalon, balonm, alonmo, lonmon, onmond, nmonda, monday, ondayk, ndaykn, daykno, ayknoc, yknock, knocke, nocken, ockeng, ckengl, kengla, englan, ngland, glando, landou, andout, ndouto, doutof, outofe, utofeu, tofeur, ofeuro, feuro2, euro20, uro201, ro2016, o2016f, 2016fi, 016fin, 16fina, 6final

Після обчислення поліноміальних хешів визначено, що потужність множини перетину множин хешів *n*-грамів рівна 8, а потужність їх об’єднання дорівнює 188. Таким чином, схожість цих двох текстів, відповідно до результату алгоритму, дорівнює 0,043.

Тестування розробленого алгоритму

Проведено тестування розробленого алгоритму. Параметри, що використовувались для тестування програмного забезпечення, такі:

1. Коефіцієнти *a* під час обчислення поліноміального хешу визначались як номер цього символу в таблиці кодів ASCII.
2. Порівнювались англомовні тексти.
3. В якості коефіцієнта *p* для обчислення поліноміального хешу обрано число 257 (воно відповідає критеріям, зазначеним у статті вище — є простим та більшим за 256 — кількість символів у таблиці ASCII.
3. Для проведення стемінгу використовувався алгоритм стемінгу Портера.
4. Довжина кожного *n*-грама дорівнювала 6.
5. Для тестування використано 10 англомовних текстів. 5 із них мали однакову тематику, решта — п’ять інших різних тем.

Результати тестування показали, що в середньому схожість текстів, які відповідають однаковій тематиці, рівна 0,051 (і всі отримані схожості лежали у проміжку від 0,0496 до 0,0691), а текстів, що відповідають різній тематиці — 0,006 (проміжок отриманих значень — від 0,004 до 0,009).

Таким чином очевидно, що отримані результати дозволяють знайти поріг, із досягненням якого два тексти новин можна вважати схожими.

Висновки

Розроблено алгоритм визначення схожості текстів новин на основі поліноміального хешування. Цей алгоритм є кращим у порівнянні з деякими класичними алгоритмами (такими як, наприклад, визначення косинусного коефіцієнта, визначення коефіцієнта Жаккара у його класичному вигляді) за рахунок того, що він спеціалізується саме на порівнянні текстів, а не на порівнянні звичайних множин. Такий ефект досягається завдяки використанню операції стемінгу, прибиранню стоп-слів, використанню синтаксичних *n*-грамів. Також використання поліноміального хешування дозволило пришвидшити машинний час порівняння текстів (адже порівнювались числа, а не рядки).

На основі цього алгоритму розроблено та реалізовано відповідне програмне забезпечення. Отримані результати засвідчують коректність розробленого алгоритму.

Алгоритм може бути використаний і для порівняння більших текстів. В такому випадку доцільно використовувати не весь вектор хешів *n*-грамів, а якусь його підмножину.

Розроблений алгоритм може бути вдосконалений шляхом визначення оптимальних значень довжини *n*-грамів.

СПИСОК ВИКОРИСТАННОЇ ЛІТЕРАТУРИ

1. Singhal Amit. Modern Information Retrieval: A Brief Overview / Singhal Amit // Bulletin of the IEEE Computer Society Technical Committee on Data Engineering. — 2001. — 24 (4). — P. 35—43.

2. Матеріали курсу Data Mining, що викладався у University of Utah [Електронний ресурс]. — Режим доступу до матеріалів : http://www.cs.utah.edu/~jeffp/teaching/cs5955/L4-Jaccard+Shingle.pdf .

3. Karen Spärck Jones. A statistical interpretation of term specificity and its application in retrieval / Karen Spärck Jones // Journal of Documentation. — 2004. — No. 60. — P. 493—502.

4. Lovins Julie Beth. Development of a Stemming Algorithm / Lovins Julie Beth // Mechanical Translation and Computational Linguistics. — 2006. — No. 11. — P. 22—31.

5. All About Stop Words for Text Mining and Information Retrieval [Electronic resource] // Text Mining, Analytics & More. — Access mode: http://www.text-analytics101.com/2014/10/all-about-stop-words-for-text-mining.html .

Рекомендована кафедрою комп'ютерних наук ВНТУ

Стаття надійшла до редакції 17.06.0216

Гранік Михайло Олександрович — аспірант кафедри комп'ютерних наук, e-mail: Fcdkbear@gmail.com;
Месюра Володимир Іванович — канд. техн. наук, доцент, професор кафедри комп'ютерних наук.
Вінницький національний технічний університет, Вінниця

M. O. Granik[1]
V. I. Mesiura[1]

Algorithm of Evaluating the Similarity of the News Articles Based on Polynomial Hashing

[1]Vinnytsia National Technical University

There has been suggested the algorithm of comparing the similarity of few news articles based on polynomial hashing. The algorithm can be used for clusterization of the news articles.

Keywords: news, news comparing, polynomial hashing.

Granik Mykhailo O. — Post-Graduate Student of the Chair of Computer Sciences, e-mail: Fcdkbear@gmail.com;
Mesiura Volodymyr I. — Cand. Sc. (Eng.), Assistant Professor, Professor of the Chair of Computer Sciences

М. А. Граник[1]
В. И. Месюра[1]

Алгоритм определения похожести новостных текстов на основе полиномиального хеширования

[1]Винницкий национальный технический университет

Предложен алгоритм сравнения похожести новостных текстов на основе полиномиального хеширования. Алгоритм может быть использован для кластеризации новостных текстов.

Ключевые слова: новости, сравнение новостей, полиномиальное хеширование.

Граник Михаил Александрович — аспират кафедры компьютерных наук, e-mail: Fcdkbear@gmail.com;
Месюра Владимир Иванович — канд. техн. наук, доцент, профессор кафедры компьютерных наук