Науковий вісник НЛТУ України
Scientific Bulletin of UNFU
https://nv.nltu.edu.ua
https://doi.org/10.36930/40300300
Article received 17.09.2020 p.
Article accepted 07.09.2020 p.
UDC 004.738.5:316.[472.45+728+772]

ISSN 1994-7836 (print)
ISSN 2519-2477 (online)

@ Correspondence author
Yu. I. Hrytsiuk
yurii.i.hrytsiuk@lpnu.ua

Н. С. Феній, Ю. І. Грицюк
Національний університет "Львівська політехніка", м. Львів, Україна

АВТОМАТИЗАЦІЯ ПРОЦЕСУ КЛАСИФІКАЦІЇ ТЕКСТОВИХ НОВИН З ІНТЕРНЕТ-САЙТІВ МЕТОДАМИ НЕЙРОННОЇ МЕРЕЖІ

Спроектовано веб-додаток, який дасть змогу здійснювати класифікацію політематичних текстових новин з інтернет-сайтів у режимі онлайн, їх зберігати і редагувати, а отримані результати ставити в чергу для подальшого оброблення та використання. Проаналізовано наявні методи класифікації політематичної текстової інформації з можливістю вибору потрібного з них чи їх комбінації, які найбільш ефективно можуть задовольняти встановлені вимоги замовників до неї за різними критеріями. Визначено метод для класифікації політематичних текстових новин, робота якого розрахована на онлайн режим їх надходження з послідовним аналізом на вході множини текстових даних.

Спроектовано архітектуру веб-додатку для послідовної класифікації текстових даних у режимі онлайн та обґрунтовано його перелік необхідних функцій, які забезпечуватимуть зберігання, оброблення та перегляд текстової інформації, отриманої внаслідок аналізу інтернет-сайтів, або даних, необхідних для його роботи. Розроблено структуру організації баз даних для реалізації веб-додатку, які забезпечать надійне зберігання класифікованої інформації за різними критеріями, а також даних для авторизації та автоматизації дій користувача.

Реалізовано веб-додаток з використанням середовища розробника, обраної мови програмування, засобів реалізації та спроектованої клієнт-серверної його архітектури, функціонал якого обробляє відповідну інформацію, використовує базу даних для її зберігання та виконання подальших дій. Для ефективної роботи веб-додатку під час класифікації текстових новин передбачено різних користувачів, потреби яких доступні за оплату, яку можна здійснити відразу на ресурсі. Користувачам доступний такий функціонал веб-додатку: оброблення, зберігання, редагування текстових новин та результатів їх класифікації, авторизації та оплати додаткових функцій.

Ключові слова: text-minig; класифікація текстової інформації; нейронна мережа; навчання "з учителем"; архітектура веб-додатку; документо-орієнтована база даних.

Вступ

Під новиною розуміють оперативне інформаційне повідомлення, яке містить соціально важливу та актуальну на даний момент інформацію, що стосується певної сфери життя суспільства загалом чи окремих його громадян зокрема [51, 59]. В журналістиці – це окремий інформаційний жанр публіцистики, який характеризується стислим викладенням ключової інформації щодо певної події, яка сталася нещодавно [57, 55]. На думку Е. Бойда "Цінність новини суб'єктивна. Чим більше новина впливатиме на життя її споживачів, на їхні прибутки й емоції, тим важливішою вона буде" [3, 13, 20, 43, 45].

Зазвичай, новини на телебаченні чи радіо передають декілька разів на день, які починаються на початку години та тривають від трьох до 15 хвилин. Вони переважно описують окремі події в таких областях, як політика, економіка, наука, культура, спорт і, на завершення, прогноз погоди. Передбачається, що новини мають бути викладені максимально нейтрально і об'єктивно без окремих коментарів тих, хто їх озвучує. Вибір новин для добірки здійснює редакція телевізійного каналу чи радіостудії, які паралельно в текстовому форматі дублюють на відповідних інтернет-сайтах [3, 11, 37].

Класифікація текстової інформації, що знаходиться на різних інтернет-сайтах, – одне з основних завдань відповідних інтернет-сервісів щодо організації текстових даних, а також подальшого її зберігання у відповідних БД. Для вирішення такого завдання все частіше використовують методи інтелектуального аналізу текстів – Text Mining [1, 3, 21, 24, 26, 28, 32, 35, 49, 50, 55, 61], найчастіше – методами нейронних мереж [10, 23]. Їхня робота зводиться до встановлення прихованих закономірностей у аналізованих текстах та її класифікації [2, 9, 13, 19, 60]. Завдання ускладнюється ще й тим, що безліч викладеної інформації на інтернет-сайтах є політематичною, тобто належить одночасно до декількох категорій [44, 48]. Водночас, більшість відомих на сьогодні методів класифікації такої інформації не враховують цієї особливості, позаяк орієнтовані на знаходження певних класів, а також не можуть в послідовному режимі класифікувати вхідні текстові дані, що є їхнім істотним недоліком.

Інформація про авторів:
Феній Назарій Степанович, студент, кафедра програмного забезпечення. **Email**: nfenii02@gmail.com
Грицюк Юрій Іванович, д-р техн. наук, професор, кафедра програмного забезпечення. **Email**: yurii.i.hrytsiuk@lpnu.ua; https://orcid.org/0000-0001-8183-3466

Цитування за ДСТУ: Феній Н. С., Грицюк Ю. І.. Автоматизація процесу класифікації текстових новин з інтернет-сайтів методами нейронної мережі. Науковий вісник НЛТУ України. 2020, т. 30, № 4. С. 123–133.
Citation APA: Fenii, N. S., & Hrytsiuk, Yu. I. (2020). Automation of the process of classification of text news from internet sites by neural network methods. *Scientific Bulletin of UNFU*, *30*(4), 123–133. https://doi.org/10.36930/40300421

Отже, на даний момент практично відсутні ефективні методи, що обробляють та здійснюють класифікацію політематичних текстових даних. Саме тому для багатьох дослідників актуальним завданням є розроблення таких методів Text Mining, які вирішували би окреслені вище проблеми.

Аналіз останніх досліджень та публікацій. Різним особливостям оброблення нечіткої інформації приурочені роботи багатьох видатних дослідників, зокрема Л. Заде, Є. Мамдані, Д. Поспєлова, Ф. Розенблатта, В. Маккаллоха, М. Мінскі, Т. Кохонена та їхніх численних послідовників. Вагомий внесок у розвиток методів класифікації текстової інформації, в тому числі й нечіткої, був зроблений такими вченими, як Д. А. Вятченін, А. П. Рижов, J. C. Bezdek, T. Kohonen [30, 31], P. Ciarelli [10], R. Krishnapuram, J. Sanches [47], B. Hammer [4, 23], Y. Kim та іншими.

Наприклад, у роботах [20, 41, 40] наведено результати аналізу масиву текстових даних на підставі концепції семантичних полів, що розглядають групи лексем, об'єднаних спільним поняттям. Такі лексеми утворюють нові характеристики текстових даних, використання яких є ефективним у завданнях їхньої кластеризації [6, 12, 14, 22, 42] та класифікації [2, 9, 13, 19, 60]. Однією із поширених моделей інтелектуального аналізу текстових даних є векторна модель, в якій ці дані подають у вигляді векторів [60] у деякому фазовому просторі [58]. Базис цього простору утворюють частотні характеристики лексем.

У роботі [20] розглянута теоретико-множинна концепція семантичних полів [13, 34] у масивах текстових даних. У роботі [41] запропонована модель кластеризації текстових даних у семантичному просторі [12, 22], яка дає можливість отримувати новий структурний поділ даних за семантичними ознаками у просторі істотно меншої розмірності, ніж у просторі, утвореному лексемним складом текстової вибірки [7].

У завданнях аналізу змісту текстових даних актуальними є теорії лексичної семантики, зокрема, вчення про семантичні поля [1, 16, 25, 27, 39, 49]. Спорідненими об'єктами у комп'ютерній інформатиці є семантичні мережі, в яких відображаються змістовні зв'язки між різними концептами [42]. Одним із прикладів ієрархічно-організованої семантичної мережі можна розглядати систему WordNet, яку було розроблено у Прінстонському університеті [15]. Лексемний склад в цій системі організовано у вигляді синсетів, під якими розуміють набори лексем синонімічного ряду, які є взаємозамінними у заданих контекстах.

У роботі [17] введено поняття семантичного домена, який описує деяку семантичну область аналізу тої чи іншої теми обговорень, наприклад, економіка, політика, фізика, програмування тощо. Для розгляду алгоритмів текстової кластеризації часто використовують стандартизовані масиви текстових даних. Однією із таких колекцій є 20-Newsgroups [http://qwone.com/~jason/20Newsgroups/], яка містить колекцію приблизно 20 тисяч документів близько 20 груп новин [6, 12, 18, 26, 56]. Цю колекцію використовують у тестових завданнях інтелектуального аналізу текстових даних, зокрема у завданнях класифікації та категоризації текстових даних [6, 9, 13, 14, 22, 60].

Об'єкт дослідження – автоматизація процесу класифікації політематичних текстових новин.

Предмет дослідження – методи і засоби класифікації політематичних текстових новин з використанням нейронної мережі, що дасть змогу початково їх аналізувати і зберігати, потім редагувати і класифікувати, а також ставити в чергу отримані результати для подальшого їх оброблення та використання.

Мета дослідження – адаптація методів нейронної мережі для класифікації політематичної текстової інформації з інтернет-сайтів, яка надходить в послідовному режимі, а також автоматизація процесів її зберігання та редагування.

Для реалізації зазначеної мети потрібно вирішити такі *основні завдання дослідження*:

- проаналізувати наявні методи класифікації політематичних текстових даних;
- розробити архітектуру нечіткої ймовірнісної нейронної мережі та методу її навчання для завдань класифікації текстової інформації в режимі онлайн;
- розробити клієнт-серверну архітектуру роботи ПЗ для реалізації завдань класифікації текстової інформації в режимі онлайн;
- розробити веб-додаток для класифікації текстової інформації в режимі онлайн, а також проаналізувати якість отриманих результатів.

Наукова новизна отриманих результатів дослідження полягає в тому, що удосконалено метод класифікації політематичних текстових новин, робота якого ґрунтується на нейронній мережі зустрічного поширення з контрольованим навчанням, що характеризується поліпшеними апроксимаційними властивостями.

Практична значущість результатів дослідження полягає в тому, що розроблений веб-додаток для класифікації політематичної текстової інформації можна використати для підвищення продуктивності роботи багатьох інтернет-сервісів, що значно покращить достовірність зібраної інформації.

Результати дослідження та їх обговорення

Поряд із значним зростанням кількості політематичних текстових новин виникає потреба автоматичного оброблення їхнього вмісту – проведення аналізу, здійснення класифікації, зберігання, реферування та інше. Для вирішення цих завдань використовують відповідне ПЗ та ефективні методи оброблення текстової інформації для великої кількості вхідних даних. Дослідження багатьох науковців [8, 18, 33, 35, 52, 55] показали, що наявні методи оброблення текстової інформації придатні для малих за обсягом даних і скеровані на вузьку тематику. Тому це не влаштовує багатьох замовників такої інформації з декількома тематиками водночас, наприклад, політематичної стрічки новин.

Мета і завдання класифікації текстових даних. Зазвичай, метою класифікації політематичної текстової інформації є поділ одних і тих самих даних на різні тематики (класи). Основне ж завдання класифікації політематичних документів полягає у визначенні їх приналежності до одного або декількох класів (з визначеного набору класів) на підставі аналізу сукупності ознак, що характеризують даний документ. Класи, до яких може належати певний документ, називають релевантними для нього. Класи в розглянутому завданні не є взаємовиключними (як у традиційній постановці завдання класифікації), а можуть перетинатися та бути вкладеними.

Отже, класифікація політематичних текстових даних не є тривіальним завданням, оскільки в невеликому фраг-

менті тексту може міститися надзвичайно цінна інформація для різних її замовників. Тому проблему віднесення даних до відповідного класу не варто ігнорувати, позаяк інформація у близько розташованих класах може перетинатися і/або навіть зливатися. До такого виду даних можуть належати потоки новинної інформації в мережі Інтернет, аналітичні огляди, сформовані новинними агентствами, наукові публікації, приурочені декільком областям досліджень, причому як близьким, так і віддаленим [33]. Наприклад, штучний інтелект й інтелектуальний аналіз даних – близькі області дослідження, онтологія інжинірингу і автоматичне оброблення текстів – віддалені, медико-біологічні чи фізико-хімічні науки – різні дослідження тощо [25]. Специфіка таких прикладних завдань, тематика яких зазвичай цікавить замовників, може динамічно змінюватися, причому – за короткий проміжок часу. Тому надзвичайно актуальним напрямом проведення досліджень стає розроблення нових методів класифікації політематичних текстових даних або удосконалення наявних, які будуть мати дещо покращені можливості.

Згідно з визначенням, текстовий документ складається з текстових об'єктів (символів, слів, абзаців) та, можливо, інших об'єктів – графічних, мультимедійних тощо [6, 12, 18, 56]. Спираючись на визначення політематичного текстового документу, вважається, що він може складатися з декількох текстових частин (модулів), які можуть відповідати одній або декільком тематикам. Тому далі такий документ розділяють на неподільні частини, які потрібно розпізнавати та аналізувати на різних рівнях структури текстового документу (наприклад, абзац, параграф, розділ).

Базуючись на методології подання текстових даних TF-IDF [15, 40, 41], яка не враховує послідовність *i*-их слів у множині політематичних текстових даних $\boldsymbol{TD}(t) = \{TD_i(t), i = \overline{1,n}\}$, що підлягають обробленню, а також поданні їх ознак у вигляді векторів $\boldsymbol{x}(t) = \{x_i(t), i = \overline{1,n}\}^T$, де $x_i(t)$ – частота повторюваності *i*-го слова у *t*-му документі, має сенс у номері способу його оброблення у навчальній вибірці.

Нехай $cl_i^{(t)} = (cl_{ij}^{(t)}, j = \overline{1,L}), i = \overline{1,n}; t = \overline{1,V}$ – вектор категорій класів для текстового документа $TD_i(t)$, елементи якого можуть набувати тільки одне з двох значень – 1 чи 0, якщо документ належить класу або не належить йому відповідно. Водночас, загальну кількість класів позначимо через $\boldsymbol{L} = \{l_j, j = \overline{1,p}\}$, де p – кількість усіх класів, яким можуть належати документи. Будемо вважати, що документ завжди буде належати хоча би одному класу $\sum_j cl_j > 0$.

Вважається, що оцінку тематичної близькості документів можна визначити не для двох документів, а серед набору суміжних за темою документів, оскільки два документи можуть бути однаковими за темою та різними за контекстом у вузькоспеціалізованій тематиці. Для обчислення значення оцінки тематичної близькості документів вибирають тільки ті слова, які є найбільш специфічними для даної тематики. Ці слова визначають за результатами апроксимації тематичного оточення документу і частоти використання їхнього напряму, що залежить від тематики [15, 17]. Такий підхід застосовують для аналізу потоку документів у науковій роботі [15].

Методи класифікації політематичних текстових новин. На сьогодні існує чимала кількість методів класифікації політематичних текстових новин [9, 13, 19], але всі вони поділяються на групи за певними ознаками: наявністю навчальної вибірки, визначення класу, способу подачі інформації на вхід класифікатора, за наявністю апріорної інформації про статистичні властивості класифікатора тощо.

За способом подачі інформації на вхід класифікатора процедури аналізу текстової інформації поділяються на [3, 21, 37]:

- послідовні, коли всі тексти новин переглядають послідовно тільки один раз;
- пакетні, на вхід яких надходять усі тексти новин у одному пакеті, тому при додаванні нового тексту потрібно робити аналіз усього вмісту пакету заново.

За наявністю апріорної інформації про статистичні властивості класів процедури аналізу текстової інформації використовують такі методи [8, 18, 33, 52]:

- параметричні, в яких для отримання невідомих ознак використовують відомі функції розподілу в межах класу, а також ймовірність появи класів. Найчастіше за допомогою таких методів будують алгоритми машинного навчання;
- непараметричні, в яких єдиний клас послідовно поділяють чи об'єднують і на його підставі формують потрібну кількість класів.

Згідно з підходами до аналізу та оброблення даних, що використовують під час класифікації політематичних текстових даних, виділяють чіткі, нечіткі, ієрархічні, неієрархічні методи [35, 55].

Для автоматизації процесу класифікації текстових даних використовують метод *k*-найближчих сусідів KNN (англ. *k-Nearest Neighbor Method*) – простий непараметричний метод, де для класифікації даних у межах простору їх властивостей використовують відстані (зазвичай евклідові), пораховані до усіх інших об'єктів. Вибираються такі об'єкти, до яких відстань найменша, і вони виділяються в окремий клас. Існує нечіткий метод *k*-найближчих сусідів [14], який допомагає дещо точніше визначити класи об'єктів при застосуванні функції [9, 36], яка визначає ймовірність входження *i*-го елемента вибірки в *j*-й клас *k*-найближчого сусіда.

Отже, метод *k*-найближчих сусідів – метричний алгоритм, який часто використовують для автоматичної класифікації текстових даних [2, 9, 13, 19, 60]. Основний принцип його роботи полягає в тому, що певну текстову інформацію присвоюють тому класу, який є найбільш поширеним серед її сусідів. Водночас, найближчі сусіди з відповідної множини вибирають тільки ті, у яких класи вже визначені. Тому, ґрунтуючись на значенні *k*, здійснюють фільтрування класів і залишають тільки ті з них, які є найбільш поширеними серед сусідів. Кожна текстова інформація має скінченну кількість атрибутів (розмірностей). Передбачається, що існує певний набір текстової інформації з уже наявною класифікацією.

Останнім часом великої популярності набув метод класифікації, робота якого базується на машинному навчанні [4, 23, 30, 33, 44, 47, 56, 60], де автоматично визначаються критерії прийняття рішення, тобто відбувається навчання текстового класифікатора. Для цього з кожного класу обирається певна кількість документів, що найбільше його характеризують, тому саме їх називають навчальними даними. Також у цьому методі кожному документу потрібно вручну прописувати його

приналежність певному класу [38], внаслідок чого формується навчальна вибірка для текстового класифікатора. Прикладом роботи такого класифікатора може слугувати Баєсова класифікація.

Метод класифікації Баєса [9] ґрунтується на теоремі, згідно з якою якщо густина розподілу кожного класу є відомою, то алгоритм розподілу текстових даних класами можна описати аналітично. Такий алгоритм володіє мінімальним набором помилок, тобто він є оптимальним. Але реально густина розподілу певного класу зазвичай невідома, тому потрібно відновлювати її за створеною навчальною вибіркою. Оскільки неможливо відновити густину вибірки без похибок, то зазначений вище алгоритм перестає бути оптимальним. Однак, при роботі з малою вибіркою є можливість підлаштувати певний розподіл під визначені дані, внаслідок чого можна зіштовхнутися з ефектом перенавчання. Отже, Баєсова класифікація даних є найбільш відомою, тому її широко використовують у теорії розпізнавання образів. Водночас, вона є базою багатьох сучасних алгоритмів розподілу текстової інформації класами, які набули широкої популярності.

Штучні нейронні мережі [10, 23] широко використовують для вирішення завдань класифікації будь-яких даних, у т.ч. й текстових. Найпопулярніші методи базуються на перцептронах [46], векторах квантування, які навчаються, і розширеній ймовірнісній нейронній мережі [23]. Під перцептроном розуміють математичну або комп'ютерну модель сприйняття інформації мозком людини [10]. Він є одним з найперших прикладів застосування штучних нейронних мереж у завданнях класифікації даних. Робота одношарового перцептрона базується на простому алгоритмі навчання, тому може вирішувати тільки найпростіші завдання. У багатошаровому перцептроні вхідний сигнал проходить через декілька шарів і перетворюється на вихідний. Сигнали в нейронній мережі мають пряме поширення тільки в одному напрямку.

Векторне квантування, що навчається (англ. *Learning Vector Quantization*) – це спосіб навчання з учителем, який використовує інформацію про клас для невеликого зміщення вектора Вороного [30]. Нейронна мережа, робота якої базується на векторному квантуванні, що навчається, має одношарову архітектуру, а налаштування її синаптичних ваг відбувається у режимі навчання з учителем, при цьому присутні елементи конкуренції за типом "переможець отримує все" [23]. Основна перевага такої мережі над іншими – це простота її архітектури, невелика кількість нейронів, що входять до неї, можливість навчання онлайн, що є важливою ознакою при вирішенні завдань класифікації текстової інформації. Також не менш значущою її перевагою є те, що вибірка даних може бути надто малою.

Описані методи класифікації текстових даних мають свої недоліки. Насамперед вони не виконують двох умов, які необхідні для вирішення завдань класифікації інформації: відсутність можливості роботи в режимі онлайн з послідовною подачею текстових даних; відсутність можливості враховувати наявність класів, що перетинаються. Більшість з відомих методів можуть виконувати тільки одну з необхідних умов.

Завдання класифікації політематичних текстових даних. Розглянемо основні завдання класифікації політематичних текстових новин. Під класифікацією текстових даних розуміють процес віднесення масиву повнотекстових даних, які отримують після аналізу і пошуку їхньої внутрішньої тематичної структури, з наявністю у ньому апріорної інформації, тобто при наявності наперед визначеного рубрикатора і множини документів-зразків. За такої класифікації документи, зв'язані між собою, прагнуть бути релевантними одним і тим самим запитам, тобто нерелевантні запиту документи відокремлені від релевантних.

Завдання класифікації політематичних текстових даних у загальному вигляді можна сформулювати так: задано множину текстів природною мовою – масив політематичних текстових даних. Будемо вважати, що існує множина тематичних груп класів $\boldsymbol{C} = \{c_j, j = \overline{1, N_C}\}$, на які можна поділити масив даних, заданий в умові. Тоді, згідно з певним критерієм якості поділу даних, завдання їхньої класифікації полягає у знаходженні оптимальної скінченної множини класів $\boldsymbol{C}$, яка є невідомою.

Зазвичай, за вхідні дані приймаємо не самі тексти інформації природною мовою, а їхнє векторне подання $\vec{\boldsymbol{D}} = \{\vec{d}_j, j = \overline{1, N_D}\}$, які характеризують смисловий зміст текстовиї даних. Ознаки таких даних автоматично формуються відповідно до обраного способу подання текстової інформації й, найчастіше, є одинарними словами. Ознаки даних об'єднують в спільну множину $\boldsymbol{P} = \{p_l, l = \overline{1, N_P}\}$, а розмірністю вектору ознак кожного набору даних є N_P.

Для зниження обсягу вхідних даних, що є поширеною практикою поділу текстових даних, часто використовують метод головних компонент, внаслідок чого відбувається спрощення векторного подання $\vec{\boldsymbol{D}}$ за рахунок синонімів і омонімів. Базуючись на статичній інформації про множину текстових даних, таке спрощення тягне за собою значні обчислювальні витрати при великих обсягах вхідних даних.

Процес класифікації текстової інформації базується на аналізі тематичної схожості даних, визначення якої ґрунтується на припущенні, що геометрична близькість векторного подання $\vec{\boldsymbol{D}}$ в просторі ознак даних всього їхнього масиву означає дійсну схожість предметних областей таких даних.

Оцінка тематичної схожості даних ґрунтується на обчисленні деякої міри схожості $Sim(\vec{d}_i, \vec{d}_j)$. Часто використовуваними ступенями схожості між векторним поданням текстових даних у просторі їхніх ознак є міра, яка визначає значення косинуса між двома векторними поданнями [60]:

$$Sim(\vec{d}_i, \vec{d}_j) = \cos\left(\angle(\vec{d}_i, \vec{d}_j)\right) = \frac{\sum_{k=1}^{N_P} d_{ki} \cdot d_{kj}}{\sqrt{\sum_{k=1}^{N_P} d_{ki}^2} \cdot \sqrt{\sum_{k=1}^{N_P} d_{kj}^2}}$$

і ступеня близькості, що ґрунтується на вимірюванні відстані між векторними поданнями даних у багатовимірному просторі їхніх ознак [22, 29]:

$$Sim(\vec{d}_i, \vec{d}_j) = 1 - Dist\left(\vec{d}_i, \vec{d}_j\right), \qquad (1)$$

де $Dist(\vec{d}_i, \vec{d}_j) = \left(\sum_{k=1}^{N_P} \left|d_{ik} - d_{jk}\right|^v\right)^{1/r}$ – відстань між векторними поданнями даних; v і r – типів відстаней – параметри, які визначає користувач. Окремі випадки типів відстаней автоматичної класифікації текстових даних є

достатньо розповсюдженими [13, 19]: евклідова відстань $(v=2, r=1/2)$, квадрат евклідової відстані $(v=2,\ r=1)$, манхеттенська відстань $(v=1, r=1)$. Нагадаємо, що манхеттенська метрика – метрика, уведена Германом Мінковським. Згідно з цією метрикою, відстань між двома точками дорівнює сумі модулів різниць їх координат. Отже, результатом розв'язання задачі автоматичної класифікації текстових даних є набір класів $\boldsymbol{C}$, структура яких залежить від вибору алгоритму класифікації, що утворюють вихідні дані задачі.

Нейромережеві методи класифікації текстової інформації. На сьогодні навчання "з учителем" нейронної мережі є найбільш поширеною й очевидною парадигмою. В такому навчанні учителю відома інформація про зовнішнє середовище, яке задають у вигляді послідовності або пакету вхідних векторів *x*, а також "правильна реакція" на ці сигнали, яку подають у вигляді навчального сигналу *d*. Природним є те, що реакція ненавченої нейронної мережі *y* відрізняється від правильної реакції вчителя, внаслідок чого виникає помилка $e = d - y$. Тому, під час навчання штучної нейронної мережі необхідно налаштувати її параметри так, щоб деяка скалярна функція $E(e)$ (критерій якості) досягла свого мінімального значення. Якщо мережа повторює реакцію учителя, то можемо зробити висновок про те, що вона є навченою. Оскільки інформація про зовнішнє середовище зазвичай має нестаціонарний характер, то для забезпечення неперервного навчання використовують ті чи інші рекурентні процедури.

Також, достатньо широко розповсюджена технологія змішаного навчання, коли частину параметрів налаштовують за допомогою вчителя, а решта або їхню частину налаштовують шляхом самонавчання. В основі алгоритмів знаходяться певні правила навчання, що тісно пов'язані із наведеними парадигмами. Визначено п'ять основних правил: навчання на підставі корекції за помилкою, навчання за Больцманом чи за Хеббом, навчання пам'яті та конкурентне навчання.

Правило корекцій за помилкою – типовий випадок навчання із вчителем, при чому за допомогою певних дій оптимізації та адаптивної ідентифікації мінімізують задану цільову скалярну функцію $E(e)$. Це правило є основою для більш ніж сотні відомих на даний час алгоритмів навчання.

В основі навчання нейронної мережі за Больцманом знаходяться принципи теоретичної термодинаміки, при цьому налаштування синаптичних ваг стохастичної мережі забезпечує бажаний розподіл ймовірностей станів окремих нейронів.

Навчання пам'яті та навчання за Хеббом належить до самонавчання нейронної мережі та базується на правилі, яке свідчить про те, що якщо нейрони з обох боків синапса знаходяться у збудженому стані, то між ними зростає зв'язок і навпаки, якщо вони знаходять в різних станах, то синаптичні ваги зменшуються.

В конкурентному навчанні описуються всі парадигми, при чому особливістю цього методу є змагання нейронів вихідного шару, за правилом "winner takes all", тобто збуджується тільки один вихідний нейрон – "переможець". Найбільш яскравим прикладом нейронної мережі, що використовують дане правило, є мережа адаптивного резонансу (ART) і самоорганізовані карти (SOM).

За допомогою ймовірнісної нейронної мережі (PNN), введеної Д. Ф. Шпехтом [10, 23], можна достатньо ефективно вирішувати завдання класифікації текстових даних. В основі принципу навчання знаходиться метод "нейрони в точках даних", через що він є достатньо швидким у роботі та простим у реалізації. Існують також модифікації PNN, які призначені для оброблення текстової інформації, а відрізняються між собою наявністю елементів конкуренції у процесі навчання та можливістю корекції рецепторних полів. Але коли обсяги даних, над якими проводиться аналіз, надто великий, а вектори ознак мають достатньо велику розмірність, то завдання класифікації значно ускладнюється. Це пояснюється тим, що у всіх нейронних мережах, які навчаються за принципом "нейрони в точках даних", кількість нейронів першого прихованого шару визначається кількістю векторів-образів навчальної вибірки *N*. Через це значно знижується швидкодія алгоритму і потреби виділити місце для зберігання всіх даних, які використовують у процесі навчання. Для того, щоб зменшити вплив цього недоліку, у роботі [10] було запропоновано покращену ймовірнісну нейронну мережу (EPNN), де перший прихований шар поданий не образами, а прототипами класів, отриманими за допомогою звичайного *k*-середнього (HCM) у пакетному режимі. Оскільки у завданнях класифікації кількість можливих класів *m* зазвичай значно менше обсягу навчальної вибірки *N*, то нейронна мережа EPNN набагато краще пристосована для вирішення прикладних завдань, ніж мережа PNN.

Як і всі відомі методи, мережа EPNN має свої недоліки: можливість навчання є тільки у пакетному режимі; навчальна вибірка має бути задана наперед; чіткий результат класифікації отримують тоді, коли при обробленні текстової інформації аналізований текст за різними рівнями належності може належати відразу декільком класам, що перетинаються.

Характеристика нейро-фаззі мережі для класифікації текстових даних. У даній роботі здійснюється спроба навчання нейро-фаззі мережі в online режимі й вона призначена для класифікації текстових даних, що подані у вигляді векторів-образів, які надходять на оброблення в послідовному режимі. Архітектура такої мережі подана на рис. 1. Дана мережа містить два шари для оброблення текстової інформації: перший шар прихованих прототипів (замість першого прихованого шару образів у звичайній PNN); вихідний шар розрахунку рівня відповідностей.

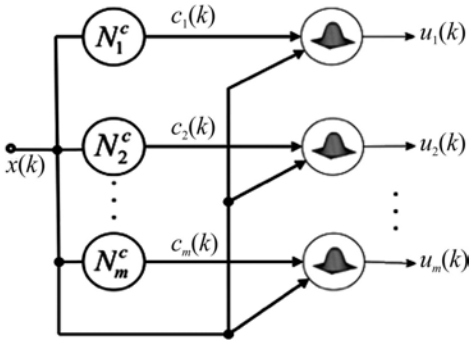

Рис. 1. Архітектура нечіткої ймовірнісної нейронної мережі (FPNN)

Вихідною інформацією для навчання нейро-фаззі мережі є класифікована послідовність векторів-образів

$\boldsymbol{x}(k)=\{x_i(k), i=\overline{1,n}\}^T \in R^n, k=\overline{1,N}$, при цьому N та кількість класів m є змінними з часом. Прототипи класів описуються векторами $\boldsymbol{C}=\{c_j=\{c_{ji}, i=\overline{1,n}\}^T, j=\overline{1,m}\}$, що підлягають визначенню, а позначення $x(k,j)$ відносить образ $\boldsymbol{x}(k)$ до j-го класу. Кожен клас містить $N_j, j=\overline{1,m}$ класифікованих образів, де $\sum_{j=1}^{m} N_j = N$.

Для розрахунку прототипів у роботі [4] використовують звичайну оцінку середнього арифметичного (НСМ), яку можна записати у рекурентній формі:

$$c_j(k)=c_j(k-1)+\frac{1}{k}\left(x(k,j)-c_j(k-1)\right), k=\overline{1,N}. \qquad (2)$$

Вираз (2), записаний у такій формі, відповідає правилу Т. Кохонена [31], з параметром кроку $\eta(k)=1/k$, що відповідає умові стохастичної апроксимації.

Оскільки у прикладних завданнях прототипи класів можуть дрейфувати в часі, то замість рекурентного виразу (2) можна використовувати експоненційно зважене середнє:

$$c_j(k-1)=\alpha\cdot c_j(k-1)+(1-\alpha)\cdot\left(x(k,j)-c_j(k-1)\right),\ 0<\alpha<1,$$

або таку адаптивну процедуру [47]:

$$\begin{cases} c_j(k)=c_j(k-1)+\eta(k)\cdot\left(x(k,j)-c_j(k-1)\right), k=\overline{1,N}; \\ \eta(k)=r^{-1}(k),\ r(k)=\alpha\cdot r(k-1)+\left\|x(k,j)\right\|^2, 0\le\alpha\le 1, \end{cases}$$

яка при $\alpha=1$ може задовольняти умови А. Дворецького [10]. Вихідний шар мережі оцінює рівень належності некласифікованих спостережень $x(k:k>N)$ до сформованих класів з прототипами $c_j(N)$ за допомогою такого виразу:

$$u_j(k)=\frac{\left\|x(k)-c_j(N)\right\|^{-2}}{\sum_{l=1}^{m}\left\|x(k)-c_l(N)\right\|^{-2}}, k=\overline{1,N}, \qquad (3)$$

який є основою ймовірнісної процедури нечіткої класифікації та відомий як метод С-середніх (FCM). Отже, у процесі навчання одночасно використовуються як чіткі, так і нечіткі процедури (НСМ і FCM). Перепишемо вираз (3) в такому вигляді:

$$u_j(k)=\frac{1}{1+\left\|x(k)-c_j(N)\right\|^2\sum_{\substack{l=1\\ l\ne j}}^{m}\left\|x(k)-c_l(N)\right\|^{-2}}, k=\overline{1,N}. \qquad (4)$$

Можна побачити, що вираз (4) є ядерною функцією активації [36], а саме

$$u_j(k)=\frac{1}{1+\frac{\left\|x(k)-c_j(N)\right\|^2}{\sigma_j^2}}, k=\overline{1,N}$$

з параметром ширини рецепторного поля, що в процесі класифікації встановлюється автоматично:

$$0\le\sigma_j^2=\left(\sum_{\substack{l=1\\ l\ne j}}^{m}\left\|x(k)-c_l(N)\right\|^{-2}\right)^{-1}\le\frac{4}{m-1}$$

Оскільки вирази (3) і (4) належать до нечіткої ймовірнісної класифікації, тобто виконується така умова:

$$\sum_{j=1}^{m} u_j(k)=1$$

то ситуація, при якій $u_j(k)=m^{-1}\forall j,$ означає, що спостереження $x(k)$ має таке саме відношення до кожного з класів, ймовірність чого є достатньо малою, або не має відношення ні до одного з них. У даному випадку можна збільшити на одиницю кількість класів $m+1$, позначивши $x(k)$ як навчальний прототип нового класу. Якщо буде визначено, що для p класів $p<m$ ступінь належності $u_j(k)$ виявиться меншим за m^{-1}, то спостереження $x(k)$ не належить цим класам і потрібно за допомогою рекурентного виразу (1) обчислити ступінь належності, змінивши верхній індекс суми в знаменнику на $m-p$.

Для вилучення можливих p класів, що не містять спостережень $x(k)$, можна використовувати процедуру, що базується на *V*-критерії (англ. *Vigilance Criterion*) і перевірці такої умови

$$e^{u_j(k)}\left\|x(k)-c_j(N)\right\|\le\varepsilon,$$

де остаточне значення ε встановлюють емпірично. Зрозуміло, що при $p = m - 1$ отримуємо чіткий результат класифікації.

Особливості реалізації архітектури ПЗ. Для реалізації ПЗ було обрано клієнт-серверну архітектуру [45, 53]. Вона є обчислювальною моделлю для розроблення комп'ютеризованих систем. Ця модель базується на розподілі функцій між двома типами незалежних і автономними процесами: сервер і клієнт. Клієнтом є будь-який процес, який вимагає конкретного запиту послуги з серверного процесу. Сервер – це процес, який надає запитані послуги клієнту. Процеси клієнта і сервера можуть перебувати на одному комп'ютері або на різних комп'ютерах, пов'язаних між собою мережею.

Коли клієнтські та серверні процеси знаходяться на двох або більше незалежних комп'ютерах у мережі, то сервер може надавати послуги для більш ніж одного клієнта. Окрім цього, клієнт може запитувати послуги з декількох серверів у мережі без урахування їхнього розташування або фізичних характеристик комп'ютера, в якому знаходиться серверний процес. Мережа зв'язує сервер і клієнт разом, забезпечуючи середовище, через яке клієнти і сервер спілкуються між собою. Наведений нижче рис. 2 показує основну комп'ютерну модель клієнта-сервера [54].

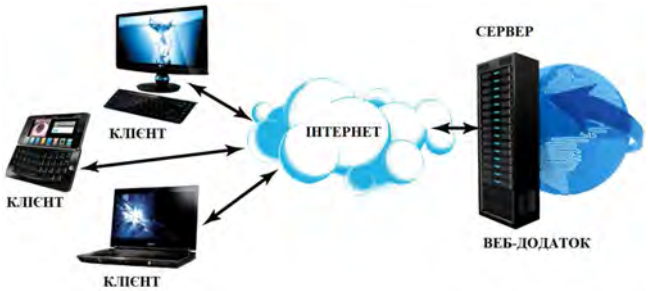

Рис. 2. Клієнт-серверна архітектурна модель роботи ПЗ

Такий тип архітектури має низку переваг серед інших моделей роботи ПЗ [53], а саме:

- забезпечення кращим обміном даних. Зазвичай, всі дані зберігаються в базі даних, тому стають доступні тільки авторизованим клієнтам. Дана властивість також захищає від несанкціонованого доступу до функцій серверу користувачам без доступу;
- ресурси спільні для різних платформ. Архітектура клієнт-сервер дає змогу додавати до ПЗ довільну кількість клієнтських частин на різних платформах: телефон, веб-додаток, комп'ютерний застосунок. При цьому не потрібно змінювати логіку роботи сервера і бази даних. Це еконо-

мить як час роботи, так і додає певні гнучкості при розробленні ПЗ;

- легке обслуговування та краща безпека. Клієнт-серверна архітектура є розподіленою моделлю і тому її достатньо легко замінити, оновити та імпровізувати цю архітектуру. Знову ж таки, оскільки сервери мають кращий і ефективний контроль над ресурсами, безпека, яка забезпечується цією архітектурою, також є достатньо жорсткою.

Архітектура веб-додатків в поєднані бібліотеки React.js і сервера майже ніколи не є завершеною, тому здебільшого є дуже складною для реалізації та супроводу. Прийнято використовувати додаткові бібліотеку React-Redux, схема роботи якої зображена на рис. 3.

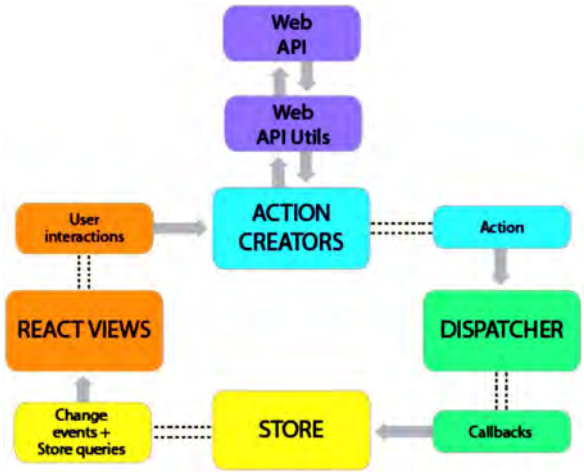

Рис. 3. Алгоритм роботи повноцінного React Native додатку

Для зручної передачі даних між елементами управління та екранами програмного продукту, бібліотека React-Redux забезпечує загальне сховище даних "Store", де програма зазвичай зберігає дані й доступається до них з будь-якого місця. Це сховище прослуховує події від "Dispatcher'а", зміни від якого будуть змінюватися в "Store" та відразу оновлюватися на елементах подання даних "View". Actions – помічники, які передають дані в "Dispatcher", а саме – назву події та дані, які необхідно обробити. Самі "View" містять елементи управління, при взаємодії користувача з якими відбувається подія і "Dispatcher" відправляє "вказівки", що робити з даними в Store. Action Creators – функції, які саме створюють дані події, тому вони можуть не тільки маніпулювати даними, але й робити зовнішні API виклики на сервери та обробляти інформацію звідти. Такий тип клієнт-серверної архітектури роботи ПЗ називають Flux (гнучким) і є не менш популярним, ніж MVC.

Сервер програмної системи будується за принципами REST (англ. *Representational State Transfer*). Оскільки дані передаються без застосування додаткових шарів, тому REST вважається менш ресурсозатратним, ніж SOAP або XML-RPC, позаяк не треба "парсити" запит, щоб зрозуміти, що він повинен зробити, і не треба переводити дані з одного формату в інший. Кожна одиниця інформації однозначно визначається глобальним ідентифікатором, таким як URL. Кожна URL водночас має строго заданий формат. Для позначення цього, які дії сервер має виконати над даними за необхідним посиланням, використовують операції CRUD (англ. *Create, Read, Update, Delete*).

За зв'язок з базою даних відповідає бібліотека mongoose, вона представляє ODM (англ. *Object Data Modelling*) – бібліотеку і працює подібно до інструментів ORM. Вона забезпечує пряме, базоване на схемі, рішення, містить вбудоване приведення до типів, валідацію, створення запитів.

Для ознайомлення із функціональними характеристиками веб-додатку розроблено відповідну діаграму прецедентів (рис. 4).

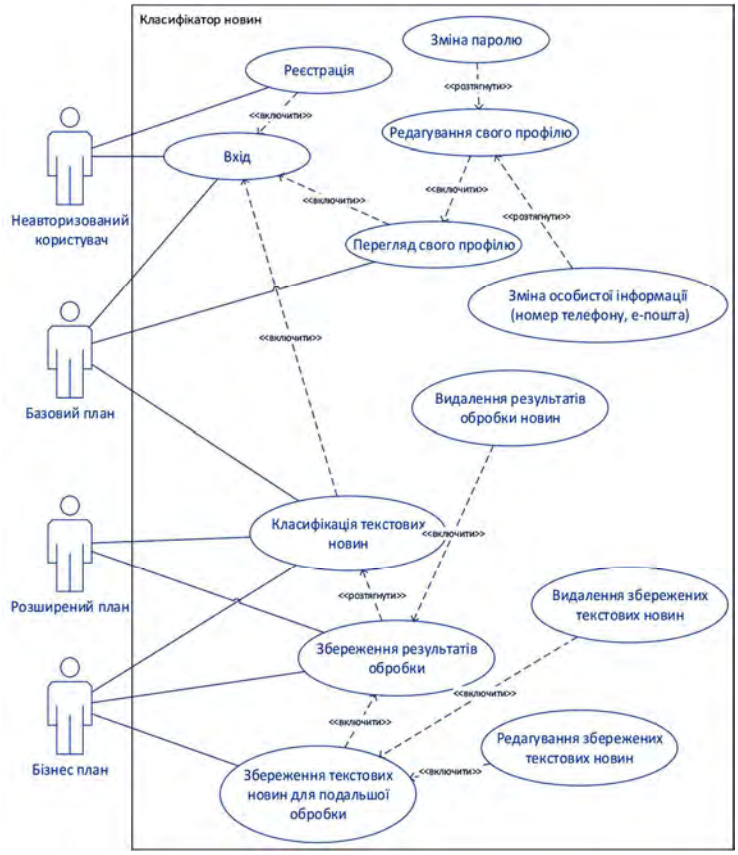

Рис. 4. Діаграма прецедентів для розроблюваного веб-додатку

Для подання статичної складової фізичної архітектури системи реалізації веб-додатку розроблено діаграму розгортання (рис. 5). На діаграмі зображено 4 частини веб-додатку, які містять клієнтську, дві серверні частини та частину бази даних. Для правильного розподілу ресурсів та коштів частина веб-додатку, яка здійснює складні обчислення, розміщена окремо та викликається тільки за потреби. Доступ до бази даних здійснюється з серверної частини веб-додатку середовища Node.js.

Як систему управління базами даних було обрано нереляційну документо-орієнтовану базу даних MongoDB. СУБД займає нішу між швидкими і масштабованими системами, що оперують даними у форматі ключ-значення, і реляційними СКБД, функціональними і зручними у формуванні запитів.

Зберігання документів в базі даних відбувається у форматі JSON, що робить її гнучкою та дає змогу зберігати великі обсяги даних. Зазвичай, нереляційні бази даних не потребують цього ж обсягу підготовчих дій, які часто необхідні для реляційних баз, що значно пришвидшує розроблення ПЗ. В цьому можна побачити велику перевагу її використання для створення програмного продукту порівняно з реляційними базами даних [5]. Модель фізичної моделі бази даних веб-сервісу створено за допомогою онлайн веб-додатку Lucidchart та зображено на рис. 6.

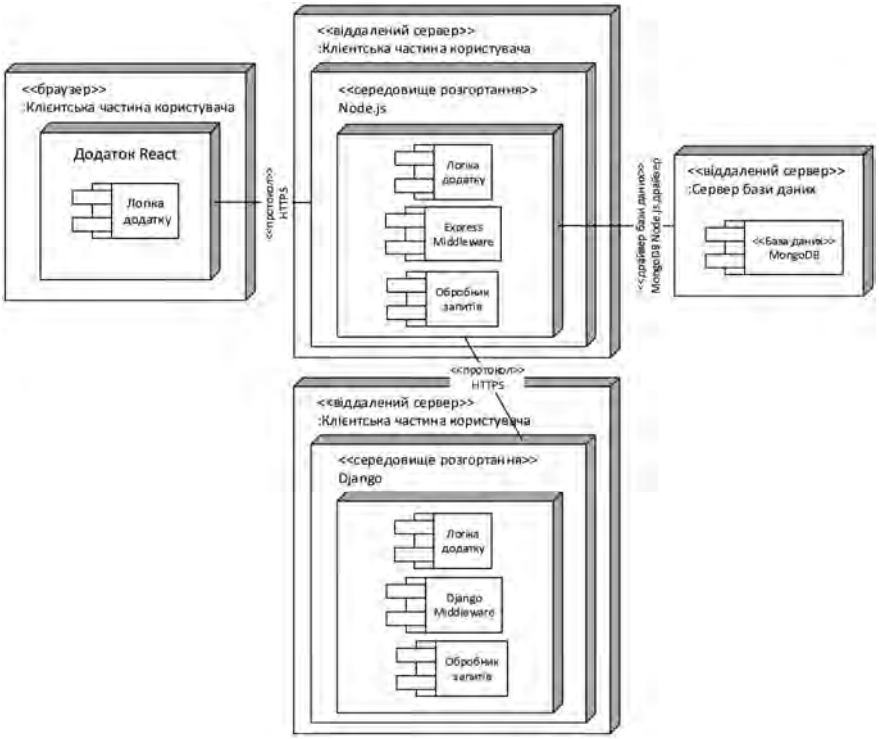

Рис. 5. Діаграма розгортання архітектури веб-додатку

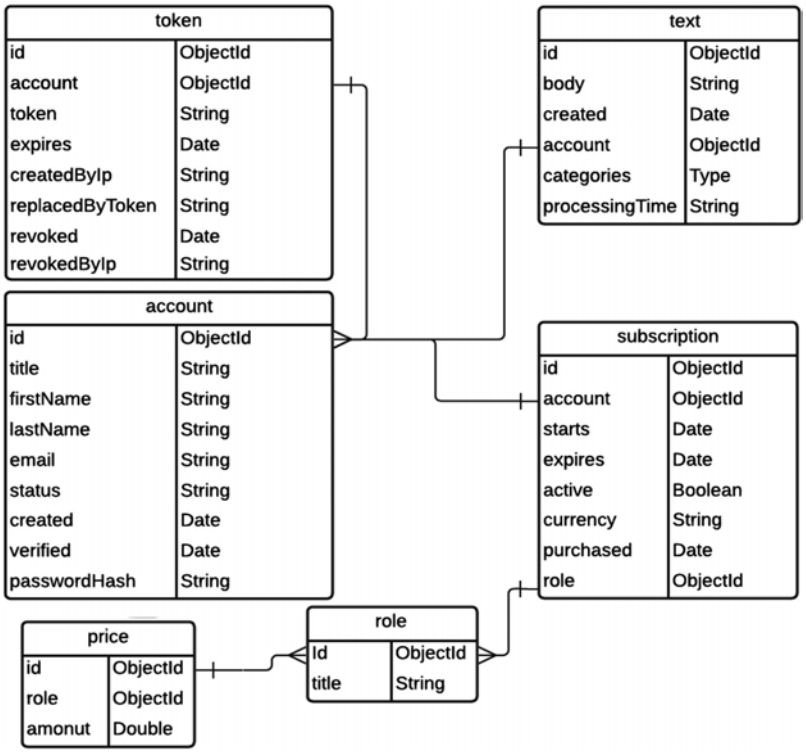

Рис. 6. Фізична модель бази даних

Веб-додаток реалізовано за допомогою системи інтегрованого розроблення Intellij Idea згідно з спроектованою архітектурою роботи ПЗ, зображеної на діаграмах, наведених вище.

Висновки

Розроблено веб-додаток, за допомогою якого можна здійснювати класифікацію політематичних текстових новин в режимі онлайн, їх зберігати й редагувати, а також ставити їх у чергу для подальшого оброблення. За результатами виконаної роботи можна зробити такі основні висновки:

1. Проаналізовано наявні методи класифікації текстових новин для вибору найоптимальнішого методу чи їхньої комбінації, які найбільш ефективно можуть задовольняти всі поставлені вимоги щодо розроблення веб-ресурсу для класифікації політематичної текстової інформації.
2. Визначено метод для класифікації політематичних текстових новин, який може працювати в режимі онлайн та на вхід послідовно отримувати множину текстових даних.
3. Спроектовано клієнт-серверну архітектуру веб-додатку для послідовної класифікації текстових новин в режимі онлайн та запропоновано набір додаткових функції, які дають змогу зберігати, обробляти та переглядати текстову інформацію, отриману внаслідок роботи веб-додатку або необхідної для його роботи.
4. Розроблено структуру організації баз даних для реалізації веб-додатку, що забезпечить надійне зберігання даних, необхідних для авторизації та автоматизації дій користувача.
5. Реалізовано веб-додаток з використанням середовища розроблення, обраної мови програмування, засобів і спроектованої клієнт-серверної архітектури його роботи. Додаток обробляє інформацію та використовує базу даних для її зберігання та подальшої роботи. Для поділу функціоналу веб-додатку передбачено три класи користувачів, можливості яких доступні за оплату, яку можна здійснити відразу на ресурсі. Користувачам доступний такий функціонал: оброблення, зберігання, редагування текстових новин та результатів їх класифікації, авторизації та оплата додаткових функцій.

References

1. Abdessalem, W. K. B., & Amdouni, S. (2011). E-recruiting support system based on Text Mining methods. *International Journal of Knowledge and Learning*, *7*(3), 220–232. https://doi.org/10.1504/IJKL.2011.044542
2. Aggarwal, C. C., & Zhai, C. (2012). A survey of text classification algorithms. In *Aggarwal, C. C., Zhai, C. (Eds.). Mining text data*, (pp. 163–222). New York, NY: Springer. https://doi.org/10.1007/978-1-4614-3223-4_6
3. Ananiadou, S., Rea, B., Okazaki, N., Procter, R., & Thomas, J. (2009). Supporting systematic reviews using Text Mining. *Social Science Computer Review*, *27*(4), 509–523. https://doi.org/10.1177/0894439309332293
4. Biehl, M., Ghosh, A., & Hammer, B. (2006). Learning vector quantization: The dynamics of winner-takes-all algorithm. *Neurocomputing*, 69, 660–670.
5. Brooks, C. (2014). *Enterprise NoSQL For Dummies*. Hoboken: John Wiley & Sons, Inc., 75 p. (John Wiley & Sons, Inc.).
6. Bsoul, Q., Salim, J., & Zakaria, L. Q. (2013). An intelligent document clustering approach to detect crime patterns. *Procedia Technology*, *11*, 1181–1187. https://doi.org/10.1016/j.protcy.2013.12.311
7. Buyukkokten, O., Garcia-Molina, H., & Paepcke, A. (2001). Seeing the whole in parts: Text summarization for web browsing on handheld devices. In *Proceedings of the 10th International Conference on World Wide Web*, (pp. 652–662). New York, NY: ACM. https://doi.org/10.1145/371920.372178
8. Cao, J., Xia, T., Li, J., Zhang, Y., & Tang, S. (2009). A density-based method for adaptive LDA model selection. *Journal of Neurocomputing*, *72*(7-9), 1775–1781. https://doi.org/10.1016/j.neucom.2008.06.011
9. Chen, J., Huang, H., Tian, S., & Qu, Y. (2009). Feature selection for text classification with naïve Bayes. *Expert Systems with Applications*, *36*(3, pt. 1), 5432–5435. https://doi.org/10.1016/j.eswa.2008.06.054
10. Ciarelli, P. M., & Oliveira, E. (2009). An enhanced probobalistic neural network approach applied to text classification. *Lecture Notes on Computer Science*, *5856*, 661–668. Berlin-Heidelberg: Springer-Verlag.
11. Cohen Priva, U., & Austerweil, J. L. (2015). Analyzing the history of cognition using topic models. *Cognition*, *135*, 4–9. https://doi.org/10.1016/j.cognition.2014.11.006
12. Conrad, J. G., Al-Kofahi, K., Zhao, Y., & Karypis, G. (2005). Effective document clustering for large heterogeneous law firm collections. In *Proceedings of the 10th International Conference on Artificial Intelligence and Law*, (pp. 177–187). New York, NY: ACM. https://doi.org/10.1145/1165485.1165513
13. Dave, K., Lawrence, S., & Pennock, D. M. (2003). Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th International Conference on World Wide Web*, (pp. 519–528). New York, NY: ACM. https://doi.org/10.1145/775152.775226
14. Derpanis, K. G. (2006). *K-means clustering*. Retrieved from: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.217.5155
15. Dias, G., Guillore, S., Bassano, J.-C., & Pereira, J. G. (2000). Lopes Combining linguistics with statistics for multiword term extraction: A fruitful association? *Proc. of Recherche dInformations Assistee par Ordinateur 2000* (RIAO2000). Retrieved from: www.di.ubi.pt/~ddg/publications/riao2000.pdf (Valid state of: 10.12.2014).
16. Dierdorff, E. C., & Morgeson, F. P. (2009). Effects of descriptor specificity and observability on incumbent work analysis ratings. *Personnel Psychology*, *62*(3), 601–628. https://doi.org/10.1111/j.1744-6570.2009.01151.x
17. Dittenbach, M., Rauber, A., & Merkl, D. (2002). Uncovering hierarchical structure in data using the growing hierarchical self-organizing map. *Neurocomputing*, *48*, 199–216.
18. El-Hamdouchi, A., & Willett, P. (1989). Comparison of hierarchic agglomerative clustering methods for document retrieval. *Computer Journal*, *32*(3), 220–227. https://doi.org/10.1093/comjnl/32.3.220
19. Faguo, Z., Fan, Z., Bingru, Y., & Xingang, Y. (2010). Research on short text classification algorithm based on statistics and rules. In *2010 Third International Symposium on Electronic Commerce and Security* (ISECS), (pp. 3–7). New York, NY: IEEE. https://doi.org/10.1109/ISECS.2010.9
20. Gavrilova, T. A., & Khoroshevsky, V. F. (2001). Knowledge bases of an intelligent system. St. Petersburg: Piter, 384 p. [In Russian].
21. Ghani, R., Probst, K., Liu, Y., Krema, M., & Fano, A. (2006). Text Mining for product attribute extraction. *SIGKDD Explorations Newsletter*, *8*(1), 41–48. https://doi.org/10.1145/1147234.1147241
22. Guo, Y., Li, Y., & Shao, Z. (2015). An ant colony-based text clustering system with cognitive situation dimensions. *International Journal of Computational Intelligence Systems*, *8*(1), 138–157. https://doi.org/10.1080/18756891.2014.963986
23. Hammer, B., & Villmann, T. (2002). Generalized relevance learning vector quantization. *Neural Networks*, *15*, 1059– 1068.
24. Holton, C. (2009). Identifying disgruntled employee systems fraud risk through Text Mining: A simple solution for a multi-billion dollar problem. *Decision Support Systems*, *46*(4), 853–864. https://doi.org/10.1016/j.dss.2008.11.013

25. Hu, J., Sun, X., Lo, D., & Li, B. (2015). Modeling the evolution of development topics using dynamic topic models. In *2015 IEEE 22nd International Conference on Software Analysis, Evolution and Reengineering* (SANER), (pp. 3–12). New York, NY: IEEE. https://doi.org/10.1109/SANER.2015.7081810
26. Jang, H., Song, S. K., & Myaeng, S. H. (2006). Text Mining for medical documents using a hidden Markov model. In *Proceedings of the Third Asia Conference on Information Retrieval Technology*, (pp. 553–559). Berlin, Germany: Springer-Verlag. https://doi.org/10.1007/11880592_45
27. Jolliffe, I. (2005). *Principal component analysis*. New York, NY: Wiley. Retrieved from: https://doi.org/10.1002/0470013192.bsa501
28. Jonsson, H., Nugues, P., Bach, C., & Gunnarsson, J. (2010). Text Mining of personal communication. In *2010 14th International Conference on Intelligence in Next Generation Networks* (ICIN), (pp. 1–5). New York, NY: IEEE. https://doi.org/10.1109/ICIN.2010.5640938
29. Kirkpatrick, S. A., Wofford, J. C., & Baum, J. R. (2002). Measuring motive imagery contained in the vision statement. *Leadership Quarterly*, *13*(2), 139–150. https://doi.org/10.1016/S1048-9843(02)00096-6
30. Kohonen, T. (1990). Improved version of learning vector quantization. *Proceedings of the 4th Int. Joint Conf. on Neural Networks*. San Diego: CA, 545–550.
31. Kohonen, T. (1995). *Self-Organizing Maps*. Berlin: Springer-Verlag, 362 p.
32. Korkontzelos, I., Mu, T., Restificar, A., & Ananiadou, S. (2011). Text Mining for efficient search and assisted creation of clinical trials. In *Proceedings of the ACM Fifth International Workshop on Data and Text Mining in Biomedical Informatics*, (pp. 43–50). New York, NY: ACM. https://doi.org/10.1145/2064696.2064706
33. Lan, M., Tan, C. L., Su, J., & Lu, Y. (2009). Supervised and traditional term weighting methods for automatic text categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(4), 721–735. https://doi.org/10.1109/TPAMI.2008.110
34. Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse Processes*, *25*(2-3), 259–284. https://doi.org/10.1080/01638539809545028
35. Lee, S., Baker, J., Song, J., Wetherbe, J. C. (2010). An empirical comparison of four Text Mining methods. In *43rd Hawaii International Conference on System Sciences* (HICSS), (pp. 1–10). https://doi.org/10.1109/HICSS.2010.48
36. Lewis, D. D. (1992). Feature selection and feature extraction for text categorization. In *Proceedings of the Workshop on Speech and Natural Language*, (pp. 212–217). Stroudsburg, PA: Association for Computational Linguistics. https://doi.org/10.3115/1075527.1075574
37. Loughran, Tim, & Bill McDonald. (2011). When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks. *The Journal of Finance 66*(1). Blackwell Publishing Inc: 35–65. https://doi.org/10.1111/j.1540-6261.2010.01625.x
38. Lughofer, E. (2011). *Evolving Fuzzy Systems – Methodologies and Applications*. Studies in Fuzziness and Soft Computing. Springer-Berlin, 456 p.
39. McKenny, A. F., Short, J. C., & Payne, G. T. (2013). Using computer-aided text analysis to elevate constructs: An illustration using psychological capital. *Organizational Research Methods*, *16*(1), 152–184. https://doi.org/10.1177/1094428112459910
40. Moyotl-Hernandez, E., & Jimenez-Salazar, H. (2004). Some Tests in Text Categorization using Term Selection by DTP. *Proceedings of the Fifth Mexican International Conference on Computer Science ENC04*. Colima, 161–167.
41. Moyotl-Hernandez, E., Jimenez-Salazar, H. (2004). An Analysis on Frequency of Terms for Text Categorization. *Procesamiento del lenguaje natural*, *33*, 141–146.
42. Osinski, S., & Weiss, D. (2005). A concept-driven algorithm for clustering search results. *IEEE Intelligent Systems*, *20*(3), 48–54. https://doi.org/10.1109/MIS.2005.38
43. Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, *2*(1-2), 1–135. https://doi.org/10.1561/1500000011
44. Phan, X.-H., Nguyen, L.-M., & Horiguchi, S. (2008). Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th International Conference on World Wide Web*, (pp. 91–100). New York, NY: ACM. https://doi.org/10.1145/1367497.1367510
45. Popescu, A.-M., & Etzioni, O. (2007). Extracting product features and opinions from reviews. In *Kao, A., Poteet, S. R. (Eds.). Natural language processing and text mining*, (pp. 9–28). London, UK: Springer. https://doi.org/10.1007/978-1-84628-754-1_2
46. Rosenblatt, F. (1962). *Principles of Neurodynamics*. New York: Spartan Books, 237 p.
47. Sanches, J. S., & Marques, A. I. (2006). An LVQ-based adaptive algorithm for learning from very small codebooks. *Neurocomputing*, *69*, 922–927.
48. Short, J. C., Broberg, J. C., Cogliser, C. C., & Brigham, K. H. (2010). Construct validation using computer-aided text analysis (CATA): An illustration using entrepreneurial orientation. *Organizational Research Methods*, *13*(2), 320–347. https://doi.org/10.1177/1094428109335949
49. Silge, Julia, & David Robinson. (2016). tidytext: Text Mining and Analysis Using Tidy Data Principles in R. *JOSS 1*(3). The Open Journal. https://doi.org/10.21105/joss.00037
50. Singh, N., Hu, C., & Roehl, W. S. (2007). Text Mining a decade of progress in hospitality human resource management research: Identifying emerging thematic development. *International Journal of Hospitality Management*, *26*(1), 131–147. https://doi.org/10.1016/j.ijhm.2005.10.002
51. Sodhi, M. S., & Son, B.-G. (2010). Content analysis of OR job advertisements to infer required skills. *Journal of the Operational Research Society*, *61*(9), 1315–1327. https://doi.org/10.1057/jors.2009.80
52. Solka, J. L. (2008). Text data mining: Theory and methods. *Statistics Surveys*, *2*, 94–112. https://doi.org/10.1214/07-SS016
53. Song, F., Liu, S., & Yang, J. (2005). A comparative study on text representation schemes in text categorization. *Pattern Analysis and Applications*, *8*(1-2), 199–209. https://doi.org/10.1007/s10044-005-0256-3
54. Subhash, C. Ya. (2009). *An Introduction to Client Server Computing*. New Delhi: New Age International (P) Ltd., Publishers, 213 p. (New Age International (P) Ltd., Publishers).
55. Vladimer B. Kobayashi, Stefan T. Mol, Hannah A. Berkers, Gábor Kismihók, Deanne N. Den Hartog. (2017). Text Mining in Organizational Research. Organizational Research Methods, 21(3), 733–765. https://doi.org/10.1177/1094428117722619
56. Vo, D.-T., & Ock, C.-Y. (2015). Learning to classify short text from scientific documents using topic models with various types of knowledge. *Expert Systems with Applications*, *42*(3), 1684–1698. https://doi.org/10.1016/j.eswa.2014.09.031
57. Wickham, Hadley. (2014). Tidy Data. *Journal of Statistical Software*, *59*(1), 1–23. https://doi.org/10.18637/jss.v059.i10
58. Yang, Y., & Pedersen, J. O. (1997). A Comparative Study on Feature Selection in Text Categorization. *The Fourteenth International Conference on Machine Learning: Proceedings of ICML97*. San Francisco, 412–420.
59. Yarkoni, T. (2010). Personality in 100,000 words: A large-scale analysis of personality and word use among bloggers. *Journal of Research in Personality*, *44*(3), 363–373. https://doi.org/10.1016/j.jrp.2010.04.001
60. Zhang, W., Yoshida, T., & Tang, X. (2008). Text classification based on multi-word with support vector machine. *Knowledge-Based Systems*, *21*(8), 879–886. https://doi.org/10.1016/j.knosys.2008.03.044
61. Zhang, Y., Chen, M., & Liu, L. (2015). A review on Text Mining. In *2015 6th IEEE International Conference on Software Engineering and Service Science* (ICSESS), (pp. 681–685). New York, NY: IEEE. https://doi.org/10.1109/ICSESS.2015.7339149

N. S. Fenii, Yu. I. Hrytsiuk
Lviv Polytechnic National University, Lviv, Ukraine

AUTOMATION OF THE PROCESS OF CLASSIFICATION OF TEXT NEWS FROM INTERNET SITES BY NEURAL NETWORK METHODS

Web application has been developed that will allow classifying, saving and editing textual news from Internet sites online. The obtained results can be queued for further processing and use. The known methods of classification of polythematic textual information with the further possibility of choosing the desired one or their combination in order to most effectively meet the established customer requirements to select information according to various criteria, are analyzed. The most effective method for classification of polythematic textual news is determined. The method for polythematic textual news classification is defined; its work is designed for online mode of their coming with sequential analysis at the input of the textual data set. Web application architecture is designed for sequential classification of textual data online; its list of necessary functions that will provide storing, processing and viewing textual information obtained from the analysis of Internet sites or data required for its operation is substantiated. The structure of the database organization for the implementation of the web application, which will ensure reliable storage of classified information according to various criteria, as well as data for authorization and automation of user actions has been developed. The web application is implemented using the developer environment, the chosen programming language, implementation tools, and the designed client server architecture. The functionality of the web application is to process the relevant information, use the database for its storage, and perform various actions as well. For the web application to work effectively, there are various users whose demands for the service are available for a fee, which can be made directly on the resource. The web application functionality available to users is as follows: processing, storage, editing of textual news and the results of their classification, authorization and payment for additional functions.

Keywords: text-minig; classification of textual information; neural network; learning "with a teacher"; web application architecture; document-oriented database.