Хмельницький національний університет
Факультет програмування та комп'ютерних і телекомунікаційних систем
Кафедра комп'ютерних наук та інформаційних технологій

ДИПЛОМНА РОБОТА

МАГІСТРА

на тему *Інформаційна система для визначення подібності документів*

Галузь знань *12 – Інформаційні технології*
Шифр і назва галузі знань

Спеціальність *122 – Комп'ютерні науки*
Шифр і назва спеціальності

Виконав: студент *2* курсу, група *КНм-19-1* — *Р.І. Прокопов*
Підпис / Ініціали, прізвище

Керівник: *к.т.н., доцент кафедри КНІТ* — *Е.А.Манзюк*
Підпис / Ініціали, прізвище

Нормоконтроль: *к.т.н., доцент кафедри КНІТ* — *Р.О. Багрій*
Підпис / Ініціали, прізвище

До захисту допускаю:
Зав. кафедри КНІТ, д.т.н., професор — *О.В. Бармак*
Підпис / Ініціали, прізвище

7 *12* 2020 р.

Хмельницький 2020

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
Факультет *програмування та комп'ютерних і телекомунікаційних систем*
Кафедра *комп'ютерних наук та інформаційних технологій*
Освітній ступінь *магістр*
Галузь знань *12 – Інформаційні технології*
Спеціальність *122 – Комп'ютерні науки*

ЗАТВЕРДЖУЮ
Завідувач кафедри *комп'ютерних наук та інформаційних технологій*
_____
(підпис)
д.т.н., професор О.В. Бармак
«*7*» *09* *2020* року

ЗАВДАННЯ
НА ДИПЛОМНУ РОБОТУ МАГІСТРА

1. Тема дипломної роботи магістра: *«Інформаційна система для визначення подібності документів»*

2. Завдання видано студенту *Прокопов Роман Ігорович*
(прізвище, ім'я, по батькові)

3. Керівник роботи *к.т.н., доцент Манзюк Едуард Андрійович*
(прізвище, ім'я, по батькові)

4. Затверджені наказом університету від «*3*» *09* *2020* р. № *22*

5. Зміст пояснювальної записки (перелік задач) та вихідні дані:

Мета роботи – розробка інформаційної технології для визначення подібності документів, та створення програмного комплексу для визначення подібності документів. Для досягнення мети необхідно дослідити існуючі підходи по напрямку визначення подібності текстовихдокументів, а також створити відповідну інформаційну систему й дослідити її ефективність. Об'єкт дослідження - нові підходи до вирішення задачі визначення подібності документів на підставі їх кластиризації. Предмет дослідження - моделі, методи, підходи та засоби для визначення подібності документів.

Реферат

Дипломна робота магістра присвячена розробці інформаційної технології для визначення подібності документів.

Актуальність теми. Щорічно у світі публікуються мільйони наукових статей. Навіть у вузькоспеціалізованих галузях науки переглядати весь обсяг інформації практично неможливо.

Крім цього в кожного дослідника за роки його роботи утворюється картотека бібліографічних описів статей, книг і т.д., що представляють для нього інтерес. Основний критерій їх відбору - особисті інтереси вченого. У цей час такі картотеки зберігаються, як правило, на електронних носіях. Це дозволяє організовувати інтегровані картотеки шляхом об'єднання ресурсів спільно працюючих дослідників.

Таким чином, виникає завдання автоматизації процесу відбору публікацій з електронних баз даних, які можуть становити інтерес для конкретного дослідника або групи спільно працюючих дослідників. Для знаходження потрібної статті дослідник звертається або до реферативних журналів, або до їхніх електронних аналогів. Тому що існують досить ефективні алгоритми пошуку конкретної публікації на електронних носіях, найбільш актуальною серед проблем інформаційного пошуку на даний момент є завдання знаходження по даному документу класу схожих по змісту документів.

Мета і завдання роботи. Метою є розробка інформаційної системи для визначення подібності документів.

Для досягнення поставленої мети визначені наступні завдання:

1. дослідити існуючі методи класифікації документів;
2. розробити інформаційну систему для визначення подібності документів;
3. виконати алгоритмічну та програмну реалізацію подібності документів.

Об’єкт дослідження. Нові підходи до вирішення задачі визначення подібності документів на підставі їх класифікації.

Предмет дослідження. Моделі, методи, підходи та засоби для визначення подібності документів.

Методи дослідження**,** використані для вирішення поставлених задач: для визначення та пошуку шаблонів в потоці даних – методи класифікації та класифікації документів; для реалізації програмного продукту – методології проектування інформаційних систем та об’єктно-орієнтований підхід.

Наукова новизна одержаних результатів. У даній роботі вирішується завдання автоматизації процесу відбору публікацій з даних. Для баз даних, які можуть становити інтерес для конкретного дослідника або групи спільно працюючих дослідників. Із цією метою на першому етапі роботи досліджуються алгоритми виміру міри подібності між двома документами електронної бази даних, а також алгоритми класифікації документів, що становлять цю базу. У якості шкал для визначення міри пропонується брати атрибути документів.

Практичне значення одержаних результатів. Створена модель, яка дозволяє визначити подібність документів.

Основна проблема класифікації документів полягає в такому рознесенні документів по групах, при якому елементи кожної групи були б настільки подібні один з одним, щоб у деяких випадках можна було зневажити їхніми індивідуальними особливостями. Зокрема, робити пошук у систематизованому файлі набагато легше, чим у несистематизованому, тому що групи документів, профілі яких не мають подібності з пошуковим приписанням, не включаються в поглиблений процес пошуку. При класифікації документів важливо прийти до розумного компромісу щодо розміру даних, уникаючи як формування великого числа дуже дрібних груп (що знижує ефективність класифікації як виділення множин схожих документів), так і невеликої кількості дуже великих класів (що може викликати зменшення точності пошуку).

Апробація дипломної роботи

Основні положення і результати роботи опубліковані в збірнику наукових праць – Прокопов Р. І. Інформаційна система для визначення подібності документів / Р. І. Прокопов, Е.А. Манзюк Т. К. Скрипник // Збірник наукових праць за матеріалами Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук - 2020» Хмельницький, 2020, – С.232-234.

Структура та обсяг роботи. Дипломна робота магістра складається з завдання, реферату, змісту, вступу, 4 розділів, висновків, переліку посилань із 17 найменувань. Загальний обсяг дипломної роботи магістра становить 74 сторінок, з них 71 сторінка основного тексту. У роботі наведено 2 рисунка та 57 таблиць.

Ключові слова: документи, кластеризація, класифікація, наїіний байес.

Зміст

Вступ

У роботі вирішується завдання автоматизації процесу відбору текстових документів наукової тематики, які можуть становити інтерес для конкретного вченого-дослідника або групи спільно працюючих дослідників. У якості шкал для визначення міри пропонується брати атрибути бібліографічного опису документів (автори, ключові слова, анотація). Значення вагових коефіцієнтів у формулі для обчислення міри подібності визначаються передбачуваною апостеріорною вірогідністю даних відповідної шкали.

У якості потенційно придатних для розв'язку поставленого завдання були проаналізовано три класичні методи класифікації документів: кластеризація шляхом знаходження клік у повній матриці подоби документів, кластеризація по методу Роккіо й метод, що базується на так званому жадібному алгоритмі, а також алгоритм, заснований на використанні функції конкурентної подібності (так званої Fris-функції). У ході тестування було виявлено, що оптимальним для даного завдання є Fris-алгоритм, хоча прийнятні результати дає й жадібний алгоритм.

Щорічно у світі публікуються мільйони наукових статей. Навіть у вузькоспеціалізованих галузях науки переглядати весь обсяг інформації практично неможливо. Внаслідок цього широке поширення одержали електронні носії інформації про нові наукові публікації, зокрема:

- бази даних Реферативних журналів;
- бази даних «Current Contents»;
- спеціалізовані мережні бази даних типу Zentralblatt MATH.

Крім цього в кожного дослідника за роки його роботи утворюється картотека бібліографічних описів статей, книг і т.д., що представляють для нього інтерес. Основний критерій їх відбору - особисті інтереси вченого. У цей час такі картотеки зберігаються, як правило, на електронних носіях. Це дозволяє

організовувати інтегровані картотеки шляхом об'єднання ресурсів спільно працюючих дослідників.

Актуальність теми. Щорічно у світі публікуються мільйони наукових статей. Навіть у вузькоспеціалізованих галузях науки переглядати весь обсяг інформації практично неможливо.

Крім цього в кожного дослідника за роки його роботи утворюється картотека бібліографічних описів статей, книг і т.д., що представляють для нього інтерес. Основний критерій їх відбору - особисті інтереси вченого. У цей час такі картотеки зберігаються, як правило, на електронних носіях. Це дозволяє організовувати інтегровані картотеки шляхом об'єднання ресурсів спільно працюючих дослідників.

Таким чином, виникає завдання автоматизації процесу відбору публікацій з електронних баз даних, які можуть становити інтерес для конкретного дослідника або групи спільно працюючих дослідників. Для знаходження потрібної статті дослідник звертається або до реферативних журналів, або до їхніх електронних аналогів. Тому що існують досить ефективні алгоритми пошуку конкретної публікації на електронних носіях, те найбільш актуальної серед проблем інформаційного пошуку на даний момент є завдання знаходження по даному документу класу схожих по змісту документів.

Мета і завдання роботи. Метою є розробка інформаційної системи для визначення подібності документів.

Для досягнення поставленої мети визначені наступні завдання:

1. дослідити існуючі методи класифікації документів;

2. розробити інформаційну систему для визначення подібності документів;

3. виконати алгоритмічну та програмну реалізацію подібності документів.

Об’єкт дослідження. Нові підходи до вирішення задачі визначення подібності документів на підставі їх класифікації.

Предмет дослідження. Моделі, методи, підходи та засоби для визначення подібності документів.

Методи дослідження, використані для вирішення поставлених задач: для визначення та пошуку шаблонів в потоці даних – методи класифікації та класифікації документів; для реалізації програмного продукту – методології проектування інформаційних систем та об’єктно-орієнтований підхід.

Наукова новизна одержаних результатів. У даній роботі вирішується завдання автоматизації процесу відбору публікацій з даних. Для баз даних, які можуть становити інтерес для конкретного дослідника або групи спільно працюючих дослідників. Із цією метою на першому етапі роботи досліджуються алгоритми виміру міри подібності між двома документами електронної бази даних, а також алгоритми класифікації документів, що становлять цю базу. У якості шкал для визначення міри пропонується брати атрибути бібліографічного опису документів.

Практичне значення одержаних результатів. Створена модель, яка дозволяє визначити подібність документів.

Основна проблема класифікації документів полягає в такому рознесенні документів по групах, при якому елементи кожної групи були б настільки подібні один з одним, щоб у деяких випадках можна було зневажити їхніми індивідуальними особливостями. Зокрема, робити пошук у систематизованому файлі набагато легше, чим у несистематизованому, тому що групи документів, профілі яких не мають подібності з пошуковим приписанням, не включаються в поглиблений процес пошуку. При класифікації документів важливо прийти до розумного компромісу щодо розміру кластерів, уникаючи як формування великого числа дуже дрібних кластерів (що знижує ефективність класифікації як виділення множин схожих документів), так і невеликої кількості дуже великих класів (що може викликати зменшення точності пошуку).

Розділ 1.
Предметна область

1.1 Загальні відомості про текстовий аналіз

Основним експериментальним матеріалом, використаному в роботі є документи, що містять інформацію відносно проведених заходів і новин у науковій сфері, а саме: оголошення про самі конкурси, загальні відомості в даній області (наприклад, проведення конференції або повідомлення про внесені зміни в організацію фондів), звертання до учасників заходів, підведення підсумків конкурсів. У такий спосіб у якості використовуваного матеріалу в роботі розглядалися тексти короткої довжини, а завданням дослідження був інтелектуальний аналіз тексту.

«Text mining» (текстовий аналіз) є частиною більш загального розділу наукових методів «Data mining» (витяг даних, аналіз даних). «Text mining» також можна вільно охарактеризувати як процес обробки тексту для витягу інформації, яка буде корисна для конкретних цілей. У порівнянні з типом даних, збережених у базах даних, текст є неструктурованим, аморфним набором, з яким важко працювати алгоритмічно, проте, у сучасній культурі, текст є найпоширенішим засобом для офіційного обміну інформацією.

В останні роки відбувається сильний ріст обсягів даних (у тому числі й текстових) як у всесвітній павутині, так і в інституціональних репозиторіях. Саме тому важливість автоматичного витягу конкретних даних з текстів, функція яких полягає в передачі й зберіганні фактичної інформації або думок, не піддається сумніву, навіть якщо результати лише частково успішні.

Головним завданням по факту є перетворення вихідного тексту до набору даних для подальшого аналізу за допомогою алгоритмів обробки даних. Значну роль при цьому відіграє спосіб представлення оброблюваних документів, способи їх попередньо обробки, визначення необхідних заходів і вагових функцій.

Існує множина додатків текстової обробки передові дослідження, що включають, аналізу й класифікації новинних повідомлень, електронних листів, фільтрації спама, ієрархічна побудова структури топиків веб-сторінок, автоматичне створення й обробки онтології й конкурентної розвідки. Кожне із цих додатків опирається на конкретну виставу корпусів тексту й множина достатня надійних, легко масштабований, не залежать від мови алгоритмів. Обчислювальні методи аналізу більших текстових корпусів можна розділити на дві основні категорії:

- статистичні;
- лінгвістичні

Статистичні методи, як правило, будуються на базі статистичній і імовірнісної структурі й часто не беруть до уваги синтаксичну й семантичну структуру тексту. Такі методи засновані на розвитку математичної представлення тексту.

Одним із самих популярних способів можна назвати матрицю слів («bag-of-word matrix»), коли кожний документ представляється вектором, що містять частоту зустрічальності кожного слова даного документа. У більш загальному виді дана матриця є деяка множина слів, складена без обліку граматики й навіть порядку слів, але зберігаючи кратність.

Лінгвістичні методи, які найчастіше засновані на обробці природньої мови, намагаються розібрати документи на основі комп'ютерної представлення людської мови. [14] Прикладами можуть послужити алгоритми синтаксичного аналізу [8, 9, 5] і автоматичної морфологічної розмітки [4]. Такий підхід може потенційно привести до більш точної представлення тексту, що лежить в основі роботи методів, що дає дорогу широкій різноманітності додатків для обробки тексту. Наприклад, більш детальна використовувана структура тексту може привести до автоматичного виділення онтологій або забезпечити зрозумілою для машини виставою контенту.

Проведене дослідження опиралося на статистичний метод, поряд з яким використовувалася інформація про частини мови слів для відбору слів у множина термінів.

1.2 Побудова векторної моделі

Векторна модель семантики (vector space model, VSM) була представлена Солтоном в 1975 г [13]. Новизна її полягала в тому, щоб використовувати частоти слів як ключової інформації для виявлення семантичної інформації. Вистава кожного компонента корпуса як крапки в багатомірному просторі (вектора у векторному просторі) містить у собі основну ідею VSM. Тут розмірність простору дорівнює потужності множині ознак моделі. Координатами є значення цих ознак, які розраховуються певним чином для кожного документа. Наприклад, безліччю ознак можуть бути всі слова документа, а за їхні значення ухвалюватися частоти слів для конкретного документа. Семантично схожим текстовим документам відповідають близько розташовані крапки простору.

Зустрічаються три найбільш популярні види матриць [3]:

– Термін-Документ. Показує подібність між документами. Для цього кожна множина слів документа представляється вектором значень, отриманим у такий спосіб: розглянемо множину {a, a, b, c, c, c}, де букви a, b, c відповідають за деяке слово вектор, що тоді відповідає, буде мати вигляд {2, 1, 3}. Тобто на першому місці коштує частота слова a, на другому слова b і на третьому - c. Порядок елементів у множині не відіграє ролі, але при побудові векторів послідовність слів-ознак повинна бути постійна. Тоді колекцію документів можна представити матрицею, де рядка ставляться до певного терміна, а стовпці відповідають деякому документу.

– Слово-Контекст. Розглядає схожість між словами. Матриця подібна моделі термін-документ, але тут стовпцями можуть бути не обов'язково документи, але й глави, абзаци, пропозиції.

– Пара-Модель відслідковує схожість відносин. Рядками матриці є пари слів, а стовпцями - різні відносини між парами слів.

У проведеній науковий праці застосовувалася матриця термін-документ.

1.3 Лінгвістична обробка

Нехай необхідно досліджувати досить великий обсяг документів, написаних природньою мовою. Попереднім етапом до побудови VSM буде предобработка тексту, яку можна розділити на три типи [3]:

1. Токенізація – ухвалення рішення про те, що буде термінами (ознаками) і способі їх витягу з вихідного тексту
2. Нормалізація – приведення всіх слів до деякої нормальної форми за рахунок єдиного (як правило нижнього) регістру й стемування.
3. Коментування (автоматична морфологічна розмітка й синтаксичний аналіз) - створення для кожного слова мітки, що вказує на приналежність його до певної частини мови.

Токенізація не завжди являє собою просте завдання. Інструмент-Розмітник повинен знати, як працювати з пунктуацією, розпізнавати переноси, а також в окремих завданнях уміти розпізнавати такі складені терміни, як «генетичний алгоритм» або «нейронна мережа». Найчастіше використовуються різні списки «стоп-слів» для відсівання термінів з високим ступенем зустрічальності в мові й не несучих важливої інформації (приводи, союзи, займенники).

Важливість нормалізації обумовлена тим, що різні частини тексту можуть мати той самий зміст. У зв'язку із цим фактом приведення всіх слів до єдиної форми відіграє настільки значиму роль.

По-перше, усі символи приводяться до одному регістру. Однак варто враховувати, що зустрічаються ситуації, коли слова різного регістру мають різний сенс.

Прикладом може послужити абревіатура СТВ (Спеціальна теорія відносності), яка після зниження регістру буде означати числівник «сто».

Необхідно також урахувати той факт, що слово має різні морфологічні форми, які не міняють його зміст, а лише зв'язують його з іншими словами в пропозиції.

Тому заміна флективних форм у документі на “нормальні” (наприклад, називний відмінок, однина, чоловічий рід для іменників) позитивно позначається на якості класифікації. Найпростішим прикладом служить процедура стемінгу (англ. «stemming»).

Даний метод полягає в приведенні вихідного слова до деякої форми (стему) шляхом відкидання максимального по довжині закінчення, яке перебуває із заздалегідь складеного набору.

У виді того, що обсяг зазначеного набору закінчень невеликий, алгоритми стемінгу показують гарний час роботи. Недолік методу полягає у виникненні великої кількості помилок, при яких не всі словоформи приводяться до одному стему (understemming) або різні за змістом слова прирівнюються до однакової нормальної форми (overstemming). Алгоритми стемінгу вивчаються в інформатиці приблизно із другої половини 20 століття.

Дослідження показали, що стемінг, а значить і нормалізація, приводить до збільшення повноти (Recall) пошуку й одночасно зменшує його точність (Precision) [2, 16].

За рахунок того, що різні слова по факту розцінюються системою як синоніми, то зростає ступінь схожості між документами, завдяки чому перебуває більше число релевантних класу документів і повнота підвищується. Зворотний ефект полягає в тому, що при зазначеному спрощенні слово може втратити споконвічно закладений зміст, що приводить до зниження точності. Дана

проблема вирішується шляхом застосування більш складних алгоритмів, таких як лематизація або методів, заснованих на правилах словотвору (наприклад, алгоритм Портеру для англійської мови [6])

Основна ідея лематизації, яка застосовується в даній роботі за допомогою сервісу Mystem [19], полягає в застосуванні словника, де кожній флективній формі слова зазначена “нормальна” (лема). Складність полягає в складанні таких словників і достатньо великому часі пошуку потрібного слова.

Загальна схема даного етапу коментування являє собою додавання міток з інформацією про частини мови, усіляких значеннях слова й синтаксичному аналізі пропозиції, що дозволяє дозволяти випадки значеннєвої неоднозначності й визначаються залежності між словами.

1.4 Математична обробка тексту

Усі тексти пройшли лінгвістичну попередню обробку в процесі якої були токенізовані, вилучені стоп-слова, усі отримані слова лемматизовані, зведені до нижнього регістру. Наступним кроком є складання матриці частот. Потім розраховуються ваги її компонентів, тому що слова, що часто зустрічаються (з високою кратністю) несуть малий обсяг інформації. Далі наявну матрицю можна «згладити»: зменшити кількість «шумів» і нульових елементів (у проведеному дослідженні не застосовується).

1.4.1 Побудова частотної матриці

Кожний елемент матриці частот являє собою числове значення *v*, відповідне до виникнення певного події: конкретний об'єкт (термін, слово) зустрівся в певній ситуації (документі, контексті) *v* раз. У теорії побудова матриці частот є простим завданням підрахунку числа появи подій.

1.4.2 Зважування ознак

Є множина підходів для визначення ваг ознак. Загальним моментів є те, що по деякому алгоритму підраховуються для кожного слова числові «ваги», які показують наскільки інформативний даний термін з погляду розв'язку якого-небудь завдання.

Одним з найпоширеніших способів оцінки ваги терма для матриць термін-документ є TF-IDF (частота терміна × зворотна частота документа) сімейство вагових функцій (Спарк Джонс, 1972).

Формули для розрахунків TF (term frequency) I IDF (inverse document frequency):

$$TF\,\frac{n_d}{\sum_D n_i} \tag{1.1}$$

де t – яке-небудь слово в документі d;

D – множина усіх документів корпуса;

n_d і n_i – скільки раз слово з'явилося в розглянутому та і-*му* документах відповідно.

Для кожного унікального слова в межах одного корпуса документів існує єдине значення IDF. Значення IDF перебуває з наступного виразу:

$$IDF(t, D) = \log\frac{|D|}{|(d_i \ni t)|} \tag{1.2}$$

де $|D|$ – число документів у колекції;

$|(d_i \ni t)|$ – скільки всього документів містять *t.*

Таким чином, міра TF-IDF виходить перемножуванням двох співмножників:

$$TF - IDF\,(t, d, D) = TF(t, d) \times IDF(t, D) \tag{1.3}$$

Елемент одержує високу вагу, коли відповідає слову, яке часто фігурує в розглянутому документі (тобто досить велике значення TF), але цей термін

рідко зустрічається в корпусі в цілому (тобто DF мала, і, таким чином, виходить висока IDF). Врахування IDF зменшує вага найбільш частих слів.

Солтон і Бакли (1988) визначили велике сімейство TF-IDF ваг функцій і оцінювали їх за результатами розв'язку завдань пошуку інформації, демонструючи, що TF-IDF зважування може принести значні поліпшення в порівнянні зі звичайною частотою.

Іншим видом зважування, часто застосовним у комбінації з TF-IDF, є нормалізація довжини (англ. length normalization) [17]. В інформаційному пошуку, якщо подібна оптимізація відсутня, пошукові системи, як правило, мають ухил на користь більш довгих документів. Нормалізація довжини коректує цей зсув.

1.4.3 Оптимізація матриці

Найпростіший спосіб для підвищення продуктивності пошуку інформації є обмеження кількості компонентів вектора. Збереження тільки елементів, що представляють слова, що найбільше часто зустрічаються, контенту, є одним з таких способів. Евристики згладжування матриць, які засновані на властивостях вагових функцій, представлених на попередньому етапі математичної обробки, дозволяють не тільки зберегти значеннєву дискримінацію, але й підвищити продуктивність обчислень.

В 1990 р. Був знайдений спосіб поліпшити вимір подоби математичною операцією на матриці термін-документ (Y) на основі знань із лінійної алгебри, який полягає в застосуванні компактного сингулярного розкладання. Даний метод дозволяє представити Y як добуток матриць $UEVT$. Використовуючи k сингулярних чисел можна одержати матрицю рангу k максимально апроксимуючу вихідну. Тим самим зменшується розмірність, а також знижує шум і ступінь розрідженості матриці Y [10].

1.5 Постановка задачі

Метою є розробка інформаційної системи для визначення подібності документів.

Для досягнення обраної мети вирішувалися наступні завдання:

– аналіз значного обсягу даних для визначення категорій, які можуть бути цікаві користувачеві (наприклад, для кого зроблене оголошення, тип оголошення, вікова групи і т.д.) і виділення основних класів усередині категорій, наприклад, по цільовій групі: аспіранти, студенти, доктори наук, кандидати наук і ін.

– розробка тестової й навчальної колекцій на основі певних категорій і класів.

– вивчення підходів до обробки природно мови й завдання класифікації, вибір стратегії обробки даних.

– вивчення й імплементація двох алгоритмів машинного навчання, що вирішують завдання класифікації

– оцінити вплив використання різних підходів нормалізації документів і значень ключових параметрів алгоритмів, визначення кращих результатів.

У якості матеріалу по розглянутій темі були використані оголошення про конкурси. Дані були класифіковані й розмічені вручну. Усього розглядалися чотири категорії, кількість документів у яких склало:

– Категорія учасників – 492;

– Тип конкурс – 399;

– Тип оголошення – 329;

– Масштаб конкурсу – 297.

У ролі інструментарію, який дозволив написати необхідну програму для проведення дослідження, використовувалася бібліотека алгоритмів машинного навчання Weka. У ході роботи були вивчені такі загальні принципи Weka, як область застосування, які завдання можливо розв'язати за допомогою даного

пакета, доступні методи, структура вхідних даних, API (інтерфейс програмування додатків).

1.6 Висновки до розділу 1

Таким чином, виникає завдання автоматизації процесу відбору публікацій з електронних баз даних, які можуть становити інтерес для конкретного дослідника або групи спільно працюючих дослідників. Для знаходження потрібної статті дослідник звертається або до реферативних журналів, або до їхніх електронних аналогів. Тому що існують досить ефективні алгоритми пошуку конкретної публікації на електронних носіях, те найбільш актуальної серед проблем інформаційного пошуку на даний момент є завдання знаходження по даному документу класу схожих по змісту документів.

Розділ 2.
Методи дослідження

2.1 Завдання класифікації даних

Однієї із завдань машинного навчання є класифікація даних поряд із кластеризацією, регресією, ранжируванням, пошуком асоціативних правил і ін. [19]. Необхідність класифікації виникає в різних сферах людської діяльності. У самому широкому змісті даний термін може мати на увазі будь-який випадок, у якому прийнято рішення або зроблений прогноз на основі наявної інформації, а процедура класифікації є відповідний метод, який дозволяє робити подібні міркування в нових ситуаціях. У цій роботі ми будемо розглядати більш строге тлумачення. Будемо вважати, що проблема класифікації стосується побудови процедури, яка буде застосовуватися до послідовності випадків, у яких кожний новий випадок повинен бути віднесений до одному з набору визначених класів на основі спостережуваних ознак або особливостей.

Розробка процедури класифікації по набору даних, для яких класи заздалегідь відомі називається навчанням із учителем. Зазначений підхід концептуально відрізняється від неконтрольованого навчання або класифікації, у якому класи (кластери) добуваються із самих даних.

Прикладами, для яких завдання класифікації є фундаментальному, можуть послужити процедури для сортування листів на основі машинного читання поштових індексів, визначення кредитоспроможності осіб на основі фінансової й особистої інформації, а також попередній діагноз хвороби пацієнта для того, щоб вибрати негайне лікування чекаючи остаточних результатів огляду. Насправді, деякі з найбільш актуальних проблем, що виникають у науці, промисловості й торгівлі можна розглядати як завдання класифікації або пошуку розв'язку з використанням складних і найчастіше досить великих даних. Можна виділити три основні напрямки досліджень по класифікації даних: статистичні аналіз, машинне навчання й нейронні мережі. [11]

У проведеному дослідженні застосовувалися інструменти машинного навчання, що вивчає побудову алгоритмів, які можуть навчатися й робити надалі прогнози для знову отриманих даних.

2.2 Алгоритми класифікації

У даному дослідженні були використано два алгоритми машинного навчання із учителем: C4.5 і наївний байесівський метод.

2.2.1 Метод C4.5

C4.5 - це один з алгоритмів побудови дерева розв'язків, який являє собою алгоритм ID3 [7] з додатковими корисними можливостями, а саме: можлива робота із числовими атрибутами, є функція обрізки галузей, що не несуть корисної інформації, допускається неповна інформація про об'єкти (значення ознак можуть бути порожніми) і ін. Той факт, що C4.5 може працювати із числовими ознаками, дозволив застосувати C4.5 для розв'язку поставленого завдання. У пакеті Weka даний алгоритм реалізований через клас J48.

Для імплементації C4.5 вихідні дані і їх структура повинні відповідати деяким параметрам:

1. Об'єкти із предметної області повинні бути описані через кінцеве число ознак (атрибутів). Необхідно, щоб набір атрибутів залишався постійним для всіх прикладів із тренувальної вибірки. Усі ознаки обов'язково повинні мати або дискретне, або, як уже було згадано вище, числове значення.
2. Вхідні дані зобов'язано бути повними щодо класів. Тобто для кожного об'єкта у вибірці повинен бути однозначно (не допускається імовірнісна оцінка) зазначений клас, до якого він ставиться.
3. Також важливо, щоб множина класів мала кінцеве число значень.
4. Кількість ознак повинна бути суттєво більше числа класів.

Алгоритм побудови дерева прийняття розв'язків. Допустимо, що вихідні дані містять множина T *об'єктів*-прикладів, а також набір атрибутів A. C – множина класів, c_i – елемент даного множині, $i\overline{1,k}$.

Процесс побудови дерева відбувається зверху вниз, тобто спочатку перебуває корінь дерева, потім його нащадки і так далі. На початковому етапі ми маємо тільки корінь, з яким зв'язана вся множина T. Потім виділяється атрибут, який найкраще класифікує приклади (з максимальною інформаційною вигодою). По його значеннях розподіляються об'єкти вихідної множині. У такий спосіб у результаті розбивки ми одержуємо n *вузлів*-нащадків T_j, де n – число значень атрибута (у випадку числової ознаки використовуються пороги значень або діапазони).

Далі процес повторюється рекурсивно для всіх отриманих підмножин і т.д. Дана процедура переривається, якщо в знову отриманого вузла всі стосовні до нього приклади належать одному класу, тоді він стає аркушем, а даний клас вказується як розв'язок. Також відбувається, якщо на деякому кроці після поділу множині за деякою ознакою серед нащадків виявилася порожня множина (тобто жоден з об'єктів не потрапив у вузол після перевірки), те асоційований з ним вузол позначається як аркуш, а розв'язком є найбільш імовірний клас для його предка.

Класифікація нового об'єкта полягає в обході дерева починаючи з кореня. У результаті перевірок класифікатор попадає в деякий аркуш, а пов'язане з ним розв'язок вказується як клас для об'єкта.

Критерій вибору атрибута, по якому відбувається розбивка. Нехай розглядається деякий атрибут $\hat{a}$(усього їх у вибірці m штук), який ухвалює n значень $v_1, v_2 \dots v_n$. Відповідно після розподілу об'єктів по даній перевірці будуть отримані n нових вузлів T_1, T_2 … T_n. Для ухвалення рішення про виділення «кращої» перевірки для поточного множині в розпорядженні є лише інформація про розподіл класів у вихідному вузлі й отриманих нащадках по обраній перевірці.

Імовірність того, що довільно обраний об'єкт із деякої множині M буде ставитися до класу c_i розраховується по класичній формулі:

$$P = \frac{num(c_i,M)}{|M|}, \quad (2.1)$$

де $\mathrm{num}(\mathrm{c_i}, \mathrm{M})$ –кількість документів з множини M, що належать класу c_i.

Далі використовується одна з версій формули Хартлі, яка говорить, що інформаційний розмір повідомлення про яку-небудь подію безпосередньо залежить від імовірності виникнення даного події [1].

$$I = \log_2(\frac{1}{P}) \quad (2.2)$$

Тоді оцінку кількості інформації, необхідної для встановлення класу об'єкта з вихідної множині T, можна представити формулою ентропії обраного множині:

$$I = -\sum_{i=1}^{k} \frac{num(c_i,T)}{|T|} * \log_2(\frac{num(c_i,T)}{|T|}) \quad (2.3)$$

Для отриманих після розбивки по перевірці $\hat{a}$ підмножин T_j застосовується наступне вираження:

$$I_{\hat{a}}(T) = \sum_{j=1}^{n} \frac{|T_j|}{|T|} * I(T_j) \quad (2.4)$$

Тоді підсумкове значення «information gain» критерію вибору перевірки розраховується для всіх атрибутів по формулі

$$IG(\hat{a}) = I(T) - I_{\hat{a}}(T) \quad (2.5)$$

У звязку того, що ентропія збільшується з наближенням розподілу класів до равноймовірними подіям, для вузла T у якості перевірки вибирається той ознака (атрибут), яка максимізує значення даного вираження, тому що необхідно розбити елементи таким чином, щоб один із класів мав суттєво більшу ймовірність щодо інших (понизити невизначеність даних).

$$a = \mathrm{argmax}_{\hat{a}\in A} IG(\hat{a}) \quad (2.6)$$

У даній роботі всі атрибути є числовими, тому необхідно вибрати поріг значень, по якому всі елементи будуть ділитися на дві множині. Т.к. кількість значень ознаки звичайно, те можна допустити, що випадково взятий числова

ознака $\hat{a}$ ухвалює значення $\{v_1, v_2 \dots v_n\}$. Потрібно розташувати значення в порядку зростання або убування. Далі послідовно розглядається пара v_i і v_{i+1}, і їхнє середнє значення t використовується для розбивки всіх об'єктів на дві групи $T^1_{th_i}$ і $T^2_{th_i}$: приклади, у яких значення обраного атрибута більше t, і ті, у яких воно менше. Для кожної ознаки перебуває поріг, по якому виходять найбільш певні підмножини:

$$t = \text{argmax}_{\overline{i1,n-1}} IG(a, th_i) == I - \frac{\left|T^1_{th_i}\right|}{|T|} * T - \frac{|T^2_{th_i}|}{|T|} * I\left(T^2_{th_i}\right) \quad (2.7)$$

На останньому кроці знаходится безпосередньо атрибут, що дає найвище число «information gain».

$$a = \text{argmax}_{\hat{a} \in A} IG(\hat{a}, th_{\hat{a}}) \quad (2.8)$$

Основною гідністю алгоритму C4.5 є його простота, але є й ряд недоліків, а саме [12]:

– Не гарантує оптимальність розв'язку, тому що може привести лише до локальної оптимізації, тобто метод ставиться до «жадібних алгоритмів» (англ. greedy algorithm).

– Досить часто відбувається перенасичення методу: відмінні показники для об'єктів із тренувальної вибірки, але погана класифікація нових випадків. Тобто алгоритм, не навчається, а лише запам'ятовує вихідні приклади.

2.2.2 Наївний байесівський метод

Даний метод заснований на теоремі Байеса, яка полягає у формулі обчислення апостеріорної ймовірності.

$$Pc|d) = \frac{P(d|c)P(c)}{P(d)} \quad (2.9)$$

де $P(c|d)$-імовірність що документ d належить класу c, яку необхідно знайти.

$P(d|c)$ - імовірність зустріти документ d серед усіх документів класу c

$P(c)$ - безумовна ймовірність зустріти документ класу c.

$P(d)$ - безумовна ймовірність документа d у корпусі документів.

Суть методу полягає в пошуку максимуму функції апостеріорної ймовірності.

Тобто розв'язком буде найбільш імовірний клас.

$$C \operatorname{argmax}_{c \in C} \frac{P(d|c)P(c)}{P(d)} \tag{2.10}$$

Ймовірність $P(d)$ не залежить від обраного класу, тому досліджувана функція зводиться до

$$C = argmax_{c \in C}\, P(d|c)P(c) \tag{2.11}$$

«Наївність» класифікатора полягає в тому, що при розрахунках використовується апроксимація ймовірності $P(d|c)$, яка являє собою добуток умовних імовірностей усіх слів з даного документа.

$$Pd|c) \approx \prod_{i=1}^{n} P(w_i|c) \tag{2.12}$$

де *wi* відповідає деякій ознаці (слову).

Отже, одержуємо наступну формулу:

$$C\ argmax_{c \in C}(P(c) \prod_{i=1}^{n} P(w_i|c)) \tag{2.13}$$

Тпро їсти передбачається, що ймовірності слів не зв'язано один з одним, що є абсолютно невірним припущенням для природньої мови.

З формули видне, що при великому обсязі документа перемножується багато маленьких чисел. Тому цілком можлива ситуація,коли порядок отриманої ймовірності вийде за межі розрядної сітки. Щоб цього уникнути можна прологарифмировать обидві частини рівняння:

$$C\ argmax_{c \in C}(\ln P(c) + \sum_{i=1}^{n} \ln P(w_i|c)) \tag{2.14}$$

Дана формула слушна на підставі монотонності логарифмічної функції. Маленькі значення перейдуть у негативні, але їх абсолютні значення будуть значно більше, що запобіжить арифметичному переповненню. У цьому випадку взятий найбільш зустрічаємий натуральний логарифм, але при розв'язку завдання підстава логарифма ролі не відіграє.

Розрахунки $P(c)$ і $P(w_i|c)$ здійснюється по тренувальній колекції.

$$P(c) = \frac{D_c}{D} \tag{2.15}$$

де D_c - потужність класу c (кількість документів даного класу);

D – загальна кількість документів у колекції.

У випадку зважених ознак, що мають чисельні значення (наприклад, вага TF-IDF, який застосовувався в проведеному дослідженні) $P(w_i|c)$ ухвалює трохи інший вид $P(w_i = \widehat{w_i}|c)$, тобто ймовірність того, що ознака w_i прийме значення $\widehat{w_i}$ для документів класу c.

У використовуваному програмному пакеті Weka ця ймовірність розраховується двома способами, які також розглядаються при проведенні експерименту:

$$P(w_i = \widehat{w_i}|c) = g(x; \mu_c; \sigma_c), \tag{2.16}$$

де $g(x,; \mu_c; \sigma_c) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$ – функція щільності нормального розподілу, параметри μ_c и σ_c беруться із навчальної вибірки.

Другий спосіб полягає у використанні функції ядра (kernel function) [15]

У даній роботі документи представляються вектором значень по множині слів-ознак, складеному по навчальній вибірці. Тобто при класифікації нового документа слова, що раніше не зустрічаються в тестовій колекції, не враховуються. Тому необхідність згладжування Лапласа або будь-якого іншого відпадає.

Перевагами наївного байесовского класифікатора служать простота реалізації й низькі обчислювальні. Головний недолік випливає з гіпотези, що фігурує, про незалежність ознак класифікації - відносно низька якість класифікації в більшості реальних завдань.

2.3 Метрики оцінювання якості

З метою виявити найбільш удалі інструменти класифікації, такі як використовуваний алгоритм із мінливими параметрами й попередню обробку документів (нормалізація, відкидання стоп-слів), отримані результати були оцінені за наступними критеріями:

– Точність (англ. precision) – показує, яка частина від тих документів, яких класифікатор порахував відповідними до розглянутого класу дійсно йому належать.

– Повнота (англ. recall) – характеризує здатність класифікатора знаходити якнайбільше об'єктів, що ставляться до класу.

– F-міра – є об'єднанням перших двох характеристик, являє собою середнє гармонійне точності й повноти.

Таблиця 2.1 - Матриця неточностей для класу X

		Очікувалося	
		1	0
Одержали	1	tp (true positive)	fp (false positive)
	0	fn (false negative)	tn (true negative)

Тут 1 означає, що елемент належить X, 0 – не належить.

– Істинно-Позитивний (**true positive**) - класифікатор прийняв вірний розв'язок про те, що даний об'єкт(документ) ставиться до класу.

– Неправильно-Позитивний (**false positive**) - отримана некоректна інформація про приналежність документа класу X.

– Неправильно-Негативний (**false negative**) - об'єкт відповідає класу, але на виході одержали зворотний результат

– Істинно-Негативний (**true negative**) - класифікатор правильно визначив документ як не стосовний до X.

Наприклад, при оцінюванні, що вертаються пошуковою системою результатів по деякому запиту класом X є релевантні документи. Даний підхід використовувався й у проведеній роботі, який буде більш докладно розглянутий у третьому розділі.

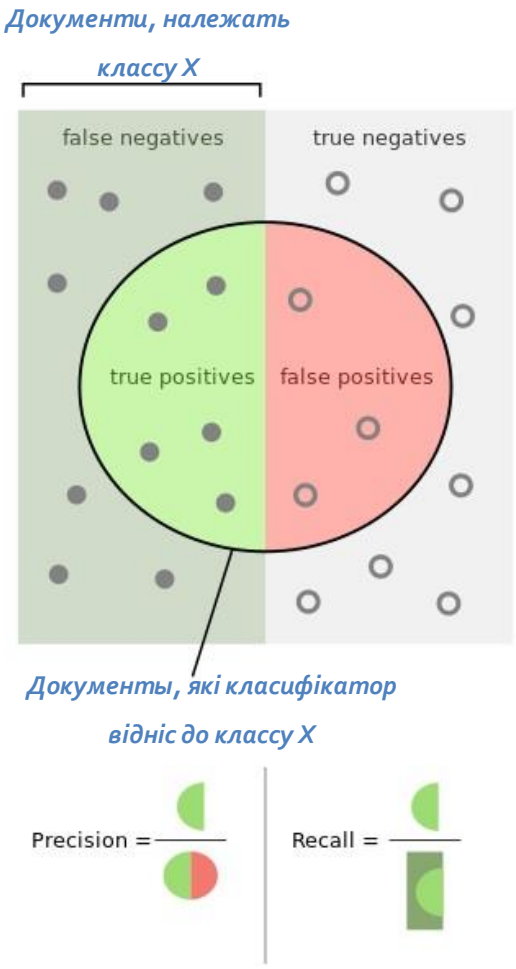


Рисунок 2.1 - Точність та повнота

Дані метрики також застосовуються в області інформаційного пошуку для оцінки якості роботи пошукових систем.

Якщо розглядати документи на приналежність деякому класу X, то всі отримані результати категоризації можна представити у вигляді таблиці, яка

називається «таблицею спряженості» або «матрицею неточностей» (confusion matrix).

Формули, по яких розраховуються метрики «точність» і «повнота» у виді введених позначень:

$$Precision = \frac{tp}{tp + fp}$$
$$Recall = \frac{tp}{tp+fn} \quad (2.17)$$
$$F\_measure = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}$$

2.4 Визначення міри подібності при класифікації текстових документів

У даній роботі вирішується завдання автоматизації процесу відбору публікацій з бібліографічних баз даних, які можуть становити інтерес для конкретного дослідника або групи спільно працюючих дослідників. Із цією метою на першому етапі роботи досліджуються алгоритми виміру міри подібності між двома документами електронної бази даних, а також алгоритми класифікації документів, що становлять цю базу. У якості шкал для визначення міри пропонується брати атрибути бібліографічного опису документів.

Під кластеризацією, розуміємо процес розбивки множині документів електронної бази на класи, при якому елементи, поєднувані в один клас, мають більша подібність, ніж елементи, що належать різним класам (у нашому випадку порівняння документів при формуванні кластерів ведеться по атрибутах їх бібліографічного опису). Кожний кластер при цьому звичайно описується за допомогою одного або декількох ідентифікаторів, називаних профілем або центроїдом. Профіль кластера може бути представлений деяким формальним об'єктом, розташованим у центрі кластера, або будь-яким представницьким об'єктом, здатним характеризувати інші об'єкти цього кластера (докладніше про

різні методи визначення центроїдів буде розказано нижче). З використанням поняття центроїда можна шукати схожі документи, порівнюючи пошукові запити спочатку із профілями кластерів, а потім, перевіряючи запису, що входять у кластери, що мають дуже близькі профілі.

Як уже згадувалося, у якості шкал для визначення міри подібності між двома документами ми використовуємо атрибути бібліографічного опису даних документів. Перелічимо основні елементи бібліографічного опису, що заносяться в картотеку документів: автори; заголовок; назва журналу або видавництва; рік виходу; тому, номер, сторінки (для публікацій у періодичних виданнях); анотація; коди класифікатора; ключові слова.

Викладемо алгоритм визначення міри подібності нового документа з документами з наявного набору (тобто особистої бібліографічної бази). Кількісна характеристика міри подібності визначається на множині документів D у такий спосіб:

$$m: D \times D \rightarrow [0, 1],$$

причому функція m у випадку повної подібності ухвалює значення 1, у випадку повної відмінності - 0. Обчислення міри подібності здійснюється по формулі виду

$$m(d_1, d_2) = \sum a_i \, m_i(d_1, d_2), \tag{2.18}$$

де i - номер елемента (атрибута) бібліографічного опису;

a_i - вагові коефіцієнти, причому $\sum a_i = 1$;

$m_i(d_1, d_2)$ - міра подібності по i-го елементу (іншими словами, по i-й шкалі).

Оскільки в описуваній ситуації практично всі шкали - номінальні, то міра подібності по i-й шкалі визначається в такий спосіб: якщо значення i- х атрибутів документів збігаються, то міра близькості рівна 1, інакше 0. При цьому необхідно враховувати, що значення атрибутів можуть бути складовими. У такому випадку $m_i = n_{i\ 1}/n_{i\ 0}$, де $n_0 = \max \{n_{i0}(d_1), n_{i0}(d_2)\}$, а $n_{i0}(d_j)$ - загальна

кількість елементів, що становлять значення і-го атрибута документа d_j, n_1 кількість співпадаючих елементів.

Зазначимо, що викладений алгоритм виміру міри подібності, може бути покладений в основу деякої експертної системи, що володіє певними продукційними правилами. Так, значення вагових коефіцієнтів a_i у формулі (2.18) може визначатися передбачуваною апостеріорною вірогідністю даних відповідної шкали. Наприклад, повне (або навіть «майже повне») збіг значень атрибута «автори» документів d_1 і d_2 більш вагомо у випадку, коли кількість значень цього атрибута в документі d_1 досить велике (у порівнянні з випадком, коли документ d_1 має всього одного автора). У такій ситуації ми можемо збільшувати значення відповідного вагового коефіцієнта у формулі (2.18) з одночасним пропорційним зменшенням інших коефіцієнтів.

2.5 Методи класифікації документів

Основна проблема класифікації документів полягає в такому рознесенні документів по групах, при якому елементи кожної групи були б настільки подібні один з одним, щоб у деяких випадках можна було зневажити їхніми індивідуальними особливостями. Зокрема, робити пошук у систематизованому файлі набагато легше, чим у несистематизованому, тому що групи документів, профілі яких не мають подібності з пошуковим приписанням, не включаються в поглиблений процес пошуку. При класифікації документів важливо прийти до розумного компромісу щодо розміру кластерів, уникаючи як формування великого числа дуже дрібних кластерів (що знижує ефективність класифікації як виділення множин схожих документів), так і невеликої кількості дуже великих класів (що може викликати зменшення точності пошуку).

Прийнято розрізняти ряд завдань класифікації: формування кластерів на основі відомостей (властивостей і характеристик) про классифицируемых об'єктах; віднесення об'єктів до сформованих кластерів або кластерів, що

перебувають у процесі формування; витяг інформації, необхідної для ідентифікації й опису класів документів. Властиво формування класів виконується звичайно на основі зіставлення векторів документів, причому клас визначається як множина усіх об'єктів, що мають досить високі значення коефіцієнта подоби. Складання характеристик класу еквівалентно побудові профілю; віднесення об'єктів до класів залежить від ступеня подоби між ідентифікаторами об'єктів і профілями класів.

У цій роботі ми досліджуємо методи класифікації, що використовують у якості критерію для порівняння тільки заздалегідь задані елементи бібліографічного опису документів, не враховуючи індивідуальні пошукові можливості даних документів і думки споживачів про їхню корисність.

У якості потенційно придатних для розв'язку поставленого завдання були проаналізовано три класичні методи класифікації документів: кластеризація шляхом знаходження клік у повній матриці подоби документів, кластеризація по методу Роккіо і метод, що базується на так званому жадібному алгоритмі, а також новий алгоритм, заснований на використанні функції конкурентної подібності (fris-аункції). Коротко викладемо суть перерахованих алгоритмів.

Процес знаходження клич заснований на побудові повної матриці подоби, за допомогою якої кожній парі документів (d_1, d_2) ставиться у відповідність коефіцієнт подоби S (d_1, d_2). Звичайно вибирається граничне значення T, і матриця подоби приводиться до бінарного виду шляхом заміни всіх коефіцієнтів подоби таких, що .S' (d_1, d_2) > T, одиницею, а всіх інших - нулем. Далі шукані класи визначаються як кличі, які можуть бути отримані з бінарного ряду подоби.

В алгоритмі Роккіо побудова матриці подоби заміняється перевіркою щільності простору деяких документів. У якості можливих центрів кластерів виступають тільки ті документи, які за результатами обчислень виявилися розташованими в щільних зонах простору. Кластеризуемий документ відносять до того класу, подоба із центроїдом якого виявилося найбільш високим.

При використанні жадібного алгоритму в матриці подоби знаходять рядок (або стовпець - матрицям симетрична), сума компонентів якої буде максимальною. Документ, відповідний до цього рядка, повідомляють центром першого кластера й включають у кластер усі документи, коефіцієнти подоби до яких більше або рівно якогось наперед заданого граничного значення. Далі викидають усі документи, що потрапили в кластер, викреслюючи з матриці відповідні рядки й стовпці, після чого процес повторяться кілька раз, поки всі документи не будуть кластеризовані.

У методі класифікації з використанням функції конкурентної подібності при визначенні міри подібності між двома документами розглядається конкурентна ситуація: розв'язок про приналежність документа d до першого кластера ухвалюється не в тому випадку, коли відстань r1 до цього кластера «мало», а коли воно менше відстані r_2 до конкуруючого кластера. Для обчислення міри конкурентної подібності, обмірюваної в абсолютній шкалі, використовується нормована величина $F_{12} = (r_2 - r/(r_2 + r_1))$, називана функцією конкурентної подібності або Fris-функцією (від Function of Rival Similarity). Зрозуміло, на первісному етапі класифікації кластерів, що коли конкурують, ще ні, доводиться працювати з деякою модифікацією (редукцією) Fris-Функції, що використовує віртуальний кластер-конкурент. Суть алгоритму полягає в тому, що з використанням скороченої Fris-функції в якості центроїдів вибираються центри локальних «згустків» розподілу документів, після чого формуються лінійно роздільні кластери.

2.6 Вибір оптимального алгоритму

У якості практичної мети застосування проаналізованих алгоритмів стояло завдання автоматизації процесу відбору публікацій з електронних баз даних, які можуть становити інтерес для конкретного дослідника або групи спільно працюючих дослідників.

Тестування алгоритмів проводилася на електронній базі даних Статтям у зазначеній базі даних крім стандартних атрибутів (назва, автор, рік видання й т.п.) приписані відповідні коди класифікатора з «Класифікації математичних сутностей». Це факт дозволив розбити всю роботу на два етапи.

Знаходження оптимального алгоритму класифікації. У якості міри на просторі документів використовується певна раніше конструкція, однак порівняння ведеться по одному єдиному атрибуту - кодам класифікатора. Оскільки збіг даних кодів для групи документів є об'єктивним критерієм збігу тематики даних документів, те такий міра можна вважати ідеальної.

Завдання міри на множині документів, яка після класифікації бази дасть результат, близький до результату з використанням міри, певної в п. 1.

Порівняння трьох класичних алгоритмів показало, що метод визначення кластерів на множині клік, отриманих з матриці подоби, показав себе малопридатним для розв'язку поставленого завдання, тому що має тенденцію до утвору великої кількості дуже дрібних груп. Алгоритм Роккіо показав трохи кращі результати: оскільки в даному методі кластеризація відбувається навколо вибіркових документів, то стало можливим поява достатня більших класів. Однак обчислена щільність простору документів виявилася такий, що й більша частина документів не ввійшла ні в один кластер.

Більш якісний результат показав жадібний алгоритм. Його використання привело до формування кластерного масиву, у якому кожний кластер містить у середньому порядку 6-10 записів (для порівняння: загальне число статей у базі даних - порядку 700). При цьому, незважаючи на необхідність побудови матриці подоби, тимчасові витрати несуттєво відрізнялися від необхідних в алгоритмі Роккіо. Таким чином, у порівнянні з методом клік і алгоритмом Роккіо жадібний алгоритм має ряд переваг:

- Відсутня проблема занадто великої кількості більших кластерів.
- Відсутня проблема занадто великої кількості дрібних кластерів.
- Неможлива поява документів, що не потрапили ні в один кластер.

– Немає проблеми визначення профілів документів, тобто центрів, навколо яких формуються кластери.

Далі було проведене порівняння Fris-алгоритму з жадібним алгоритмом. З'ясувалося, що Fris-алгоритм дає кращу точність класифікації.

На гістограммах відображений склад отриманих кластерів. По горизонтальній осі відкладені умовні номери кластерів (відповідно темам або іншим розділам класифікатора), по вертикальній - кількість документів у кластері. У якості критерію перевірки правильності віднесення публікації до кластера використовувався його код класифікатора. Якщо коди класифікатора центроїда кластера втримувалися в числі кодів класифікатора даному запису, то ми вважалися, що запис був віднесений до кластера правильно.

Як неважко помітити, величина «шуму» (відображувана у верхній частині стовпчиків) у кластерах при класифікації Fris-алгоритмом суттєво нижче, ніж у випадку жадібного алгоритму. Більше того, розбивка на кластери більш рівномірно, а відсоток одноелементних кластерів суттєво нижче.

До порівняльних недоліків Fris-алгоритму слід віднести необхідність вручну задавати число кластерів у розбивці, а також трохи більшу обчислювальну складність - O (kn^2), де до - число, що задається користувачем, кластерів, - у порівнянні з O (N_2) у жадібного алгоритму. Однак при класифікації великих баз таке збільшення складності стає не настільки істотним, до того ж для створення системи, що автоматизує процес відбору даних, кластеризацію бази даних потрібно проводити тільки один раз. Таким чином, у якості оптимального алгоритму для розв'язку завдання класифікації баз даних даних був визнаний Fris-алгоритм.

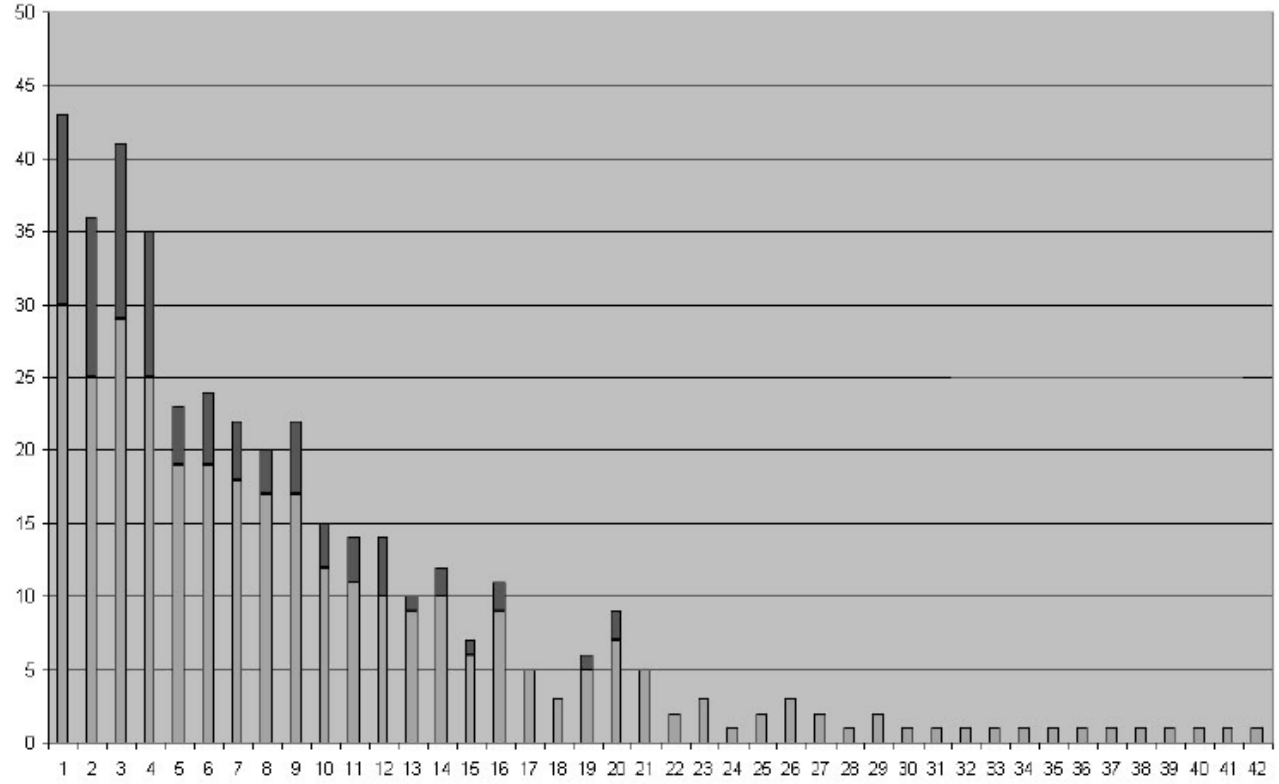


Рисунок. 2.2 - Жадібний алгоритм

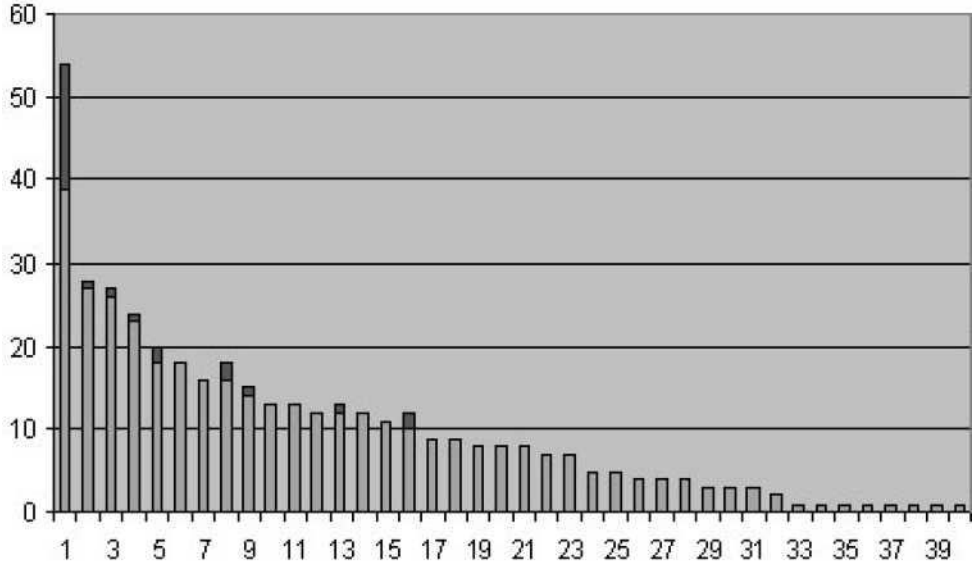


Рисунок. 2.3 – fris-алгоритм

Виявлення оптимального способу завдання міри подібності на множині документів

Для завдання міри на множині документів ми застосували формулу (1.1), де в якості шкал використовувалися наступні атрибути бібліографічного опису: автори; ключові слова; анотація.

Тому що порівняння анотацій у явному виді (тобто як текстових рядків), мабуть, безглуздо, те в якості окремої подзадачи вирішувалося питання виділення термінів із загального тексту анотацій. У даний момент доступно web-додаток [Барахнин, Куперштох,, що генерує по запиту xml-документ із переліком вхідних у даний запит математичних термінів (як джерело термінів використовується тезаурус. Таким чином, після інтеграції цього додатка в кластеризующую документи програму, дана подзадача була вичерпана, і анотації стало можливим порівнювати між собою як інші складені атрибути.

Крім того, при завданні міри був прийнятий в увагу той факт, що значення вагових коефіцієнтів у формулі (1.1) визначаються передбачуваною апостеріорною вірогідністю даних відповідної шкали й у певних випадках один з коефіцієнтів може бути збільшений із пропорційним зменшенням інших.

Для визначення вагового коефіцієнта при кожному з атрибутів була проведена кластеризація вибірок з бази даних . Нами були розглянуті вибірки різної потужності, а в якості критерію істинності застосовувався результат класифікації.

Як показав експеримент, найбільша подібність із результатом класифікації при мері, що базується на кодах класифікатора, було досягнуто шляхом введення наступних продукційних правил.

Якщо кожний з документів d_1 і d_2 має більш двох авторів і як мінімум 2/3 із числа авторів збігаються, то відповідний ваговий коефіцієнт при атрибуті «автори» ми вважаємося рівним одиниці.

Якщо кожний з документів d_1 і d_2 містить більш трьох ключових слів і як мінімум $^{3/4}$ цих слів збігаються, то відповідний ваговий коефіцієнт при атрибуті «ключові слова» ми вважаємося рівним одиниці.

Якщо кожний з документів d_1 і d_2 містить більш чотирьох термінів тезауруса в анотації і як мінімум 3/5 цих термінів збігаються, то відповідний ваговий коефіцієнт при атрибуті «ключові слова» ми вважаємося рівним одиниці.

У противному ж випадку ми вважаємося коефіцієнт при атрибуті «автори» рівним 0,2, а при атрибутах «ключові слова» і «анотація» рівним 0,4.

Цікаво відзначити, що ці правила виявилися оптимальними як для жадібного алгоритму, так і для Fris-Алгоритму.

Висновок до розділу 2

Був розроблений і протестований спосіб завдання міри подібності документів, що ґрунтується на порівнянні атрибутів бібліографічного опису даних документів.

Також було проведене дослідження різних алгоритмів класифікації документів з метою виявлення оптимального алгоритму для розбивки масиву записів електронної бази з інформацією про наукові публікації, на кластери, що містять у собі статті по подібній тематиці. Тестування алгоритмів проводилося на електронній базі даних.

Важливим аспектом у сучасному суспільстві є гонка технологій і постійний ріст темпів наукового прогресу. Розвиток існуючого потенціалу вчених, допомога в просуванні їх ідей, надання максимальне комфортних умов для проведення досліджень – усі ці питання регулярно піднімаються як окремою державою, так і світовим співтовариством у цілому. З метою їх знаходження формується величезна кількість фондів і програм, які проводять усілякі конкурси й міри з різними цільовими групами. Але встає проблема доступності даної інформації для окремого вченого. Оголошення, як правило, публікуються на сайтах організаторів, тобто дані досить розрізнені, і окремій особі складно вчасно відслідковувати нові публікації. При одержанні інформації з різних

джерел результатом буде досить великий обсяг документів, більша частина яких не буде цікава окремій особі.

Таким чином, актуальне завдання збору повідомлень у науковій сфері і їх автоматичної представлення у вигляді зручному для швидкого фасетного пошуку. Передбачається, що останнє дозволить ученому настроїти потрібні фільтри й одержати тільки той набір оголошень, який цікавий безпосередньо йому.

Одержання первинних результатів для розробки такої системи лягло в основу даної дипломної роботи, метою якої є створення інформаційної системи для визначення подібності документів.

Розділ 3

Формування даних для класифікатора

3.1 Розробка схеми класифікації

У зв’язку з тим, що в якості методу досягнення поставленої мети прийнята класифікація із учителем, першим кроком необхідно підготувати навчальну колекцію. У першу чергу була потрібна схема класифікації, яка була б найбільш близька до реальної картини сприйняття оголошень про конкурси в науковій сфері людиною. Об'єктами класифікації є текстові документи невеликого розміру. Після вивчення вихідних даних 4 категорії:

1) категорія учасників

– інше: організації, журналісти, працівники компаній, суб'єкти малого підприємництва, експерти, що вчаться шкіл/коледжів і т.п.

– доктори наук;

– кандидати наук;

– молоді вчені, дослідники, викладачі;

– молоді доктори наук;

– молоді кандидати наук;

– аспіранти;

– студенти.

2) тип конкурсу

– інше: проекти проведення різних наукових заходів (конференцій, симпозіумів, експедицій, шкіл), творчі конкурси (есе, дизайну, відеороликів і ін. форматів) і т.п.

– наукових/дослідницьких проектів, гранти на проведення досліджень

– премії й стипендії (іменні, уряду і ін.), а також конкурси вже виконаних наукових праць.

– наукова мобільність (навчання й виконання дослідницької роботи за рубежем, стажування, школи й конференції)

– інноваційних проектів, стартапів.

3) тип оголошення

– про конкурс;

– оголошення результатів;

– загальна інформація: зміни в організації різних фондів, оголошення про наукові заходи не на конкурсній основі (конференції, наукові школи й ін.);

– інформація для учасників або переможців конкурсів, керівників проектів, грантоотримувачів.

4) масштаб конкурсу

– не зазначене;

– міжнародний;

– державний;

– вузівський ;

– регіональний, міський

3.2 Організація навчальнї й тестової множин

Першим кроком складалася тестова колекція в такий спосіб: з множині оголошень послідовно відбирався й розмічався набір з 150 текстів. Послідовний вибір проводився з метою максимального наближення до реальної ситуації й текстам, які потрібно буде класифікувати. Проте, для деяких класів документів виявилася недостатня кількість (за мінімум було взято 15 об'єктів). Тому вони були доповнені документами, знайденими по ключових словах і перевіреними вручну на приналежність класу. Наприклад, для першої категорії класу «5» ключовими словами будуть: «аспірант», «аспірантура» і «phd». Аналогічним останньому способу методом розроблялася навчальна вибірка.

Споконвічно передбачалося, що один текст у деякій категорії може належати декільком класам. Як показав результат ручної розмітки, це характерно тільки для першої категорії. Для інших кількість документів із множинним спадкуванням не перевищило семи відсотків. Тобто, приміром, лише три документи з вісімдесяти, що ставляться до класу «конкурс проектів, грантів» категорії «тип конкурсу», також відповідали іншому класу цієї категорії. У результаті в тестовій і тренувальній колекції лише для класів категорії учасників були сформовано дві додаткові групи «належить» і «не належить».

Тренувальна вибірка являє собою ієрархічну систему текстових файлів, у якій на верхньому рівні розміщені папки-категорії, кожна з яких містить папки all і classes. У каталозі all зберігаються всі приклади даної категорії, файли не повторюються й мають унікальні імена, які надалі використовуються для підрахунку TF-IDF для кожного слова й побудові VSM документа. У папці classes розташований рівень папок-класів, що включають у себе (лише у випадку першої категорії) два підкласи: 1 – документи відповідають зазначеному класу, 0 – не відповідають. У каталогах 0 і 1 зібране приблизно рівне (різниця менш 5 відсотків від загального числа) кількість файлів-прикладів (імена збігаються з папкою all), щоб у класифікатора не було некоректної інформації про кількісну перевагу одного класу над іншим. Усі файли мають розширення .txt і кодування UTF-8.

Структура тестової вибірки відрізняється лише відсутністю папки all у першій категорії (тому що при розрахунках ваг слів для обраного документа враховуються лише об'єкти-приклади з навчального множині, тому зважування термів відбувається безпосередньо перед визначенням класу), а також відсутністю твердого зв'язку між підкласами 0 і 1, вони заповнюються за результатами ручного сортування файлів.

Далі наведені деякі відомості по отриманих навчальному й тестовому безлічах.

3.3 Загальні відомості по отриманих вибірках

Таблиця 3.1 - Тренувальна колекція, загальні відомості

Нормалізація	Середня кількість слів у документах	Кількість унікальних слів
Відсутні	217	53418
Mystem (усі слова)	236	12195
Mystem (підм. & прис. & дієслова)	188	8643
Mystem (підм. & прис.)	162	6897

Таблиця 3.2 - Тренувальна колекція. Кількість документів по категоріях

Категорія	Кількість документів
категорія учасників	492
тип конкурсу	399
тип оголошення	329
масштаб конкурсу	297

Таблиця 3.3 - Тестова колекція. Кількість документів по категоріях формація з отриманих класів у тестовій вибірці

Категорія	Кількість документів
категорія учасників	100
тип конкурсу	145
тип оголошення	158
масштаб конкурсу	134

Таблиця 3.4 - Перша категорія. Кількість документів по класах

Клас	Кількість документів у підкласі 1 (належить)
інше	31
доктори наук	16
кандидати наук	15
молоді вчені	39
Молоді дос.	20
молоді к.н	25
аспіранти	39
студенти	31

Таблиця 3.5 - Друга категорія. Кількість документів по класах

Клас	Кількість документів
інше	23
проекти & гранти	41
премії, стипендії & виконані роботи	37
наукова мобільність	29
Стартапи & інноваційні проекти	15

Таблиця 3.6 - Третя категорія. Кількість документів по класах

Клас	Кількість документів
оголошення про конкурс	111
оголошення результатів	16
загальна інформація	16
інформація для учасників	15

Таблиця 3.7 - Четверта категорія. Кількість документів по класах

Клас	Кількість документів
не зазначене	20
міжнародний	41
місцевий	43
внутрішнвузівський	15
міський & регіональний	15

3.4 Інформація про розміри класів у навчальній вибірці

Таблиця 3.8 - Перша категорія. Кількість документів по класах

Клас	Кількість документів у підкласі 1 (належить)
інше	168
доктори наук	47
кандидати наук	42
молоді вчені	219
молоді н.	95
молоді к.н	105
аспіранти	206
студенти	195

Таблиця 3.9 - Друга категорія. Кількість документів по класах

Клас	Кількість документів
інше	76
проекти & гранти	61
премії, стипендії & виконані роботи	75
наукова мобільність	97
Стартапи & інноваційні проекти	90

Таблиця 3.10 - Третя категорія. Кількість документів по класах

Клас	Кількість документів
оголошення про конкурс	154
оголошення результатів	63
загальна інформація	41
інформація для учасників	71

Таблиця 3.11 - Четверта категорія. Кількість документів по класах

Клас	Кількість документів
не зазначене	61
Міжнародний	74
Місцевий	61
внутрішньовузівський	45
міський & регіональний	46

Висновки по розділу 3

Порівняння класичних алгоритмів показало, що метод визначення подібності на множинах, отриманих з матриці подібності, показав себе малопридатним для розв'язку поставленого завдання, тому що має тенденцію до утворення великої кількості дуже дрібних груп. Алгоритм Роккіо показав трохи кращі результати: оскільки в даному методі кластеризація відбувається навколо вибіркових документів, то стало можливим поява достатньо великих класів.

Розділ 4
Проведення експериментів

4.1 Опис кроків обробки даних

Після того як були організовані навчальна й тестова колекція, усі документи, які в них утримуються випливало попередньо обробити й привести до одному виду. Першим кроком проводилася токенізація, потім лематизація слів і визначення частин мови з використанням сервісу Mystem [20], який показав порівняно непогані результати на корпусі текстів російською мовою - ruscorpora. Даний програмний продукт проводить морфологічний аналіз тексту російською мовою, а також є присутнім можливість побудови гіпотетичних розборів для слів, що не входять у словник.

Розглядалися 3 випадку представлення документів:

1. Усі слова у вихідному виді.
2. Усі слова після лематизації.
3. Лематизовані іменники, прикметники й дієслова.

Експеримент повинен був дозволити визначити кращий підхід до представлення документів.

Далі програма становить вектор унікальних слів корпуса документів, який буде використовуватися класифікатором у якості множині ознак. Потім для кожного слова всіх документів розглянутої категорії розраховується вага TF-IDF, тим самим одержуємо наступну виставу об'єкта вибірки (документа): вектор значень, де на і-ой позиції коштує підрахована вага TF-IDF слово, відповідне до цієї позиції вектора ознак. Були підготовлені різні набори стоп-слів: Яндексу, стандартний і розширений списки Вордстата, а також порожній список (слова не відкидалися)

4.2 Результати експериментів

Вихідним кроком для проведення експериментів стало визначення принципу вибору ознак векторної моделі документів. Для цього виставлявся мінімальний поріг довжини слова, а також використовувалися різні списки стоп-слів (Яндекс, Вордстат, розширений Вордстат) Досвідченим шляхом не було виявлено підходу, який би стабільно показував кращі результати. Показники варіювалися від класу до класу й при застосуванні різних алгоритмів класифікації. Проте вдалося встановити, нижній поріг довжини слова. Було ухвалене рішення використовувати список стоп-слів Яндексу й у якості ознак використовувати слова, довжини яких більш двох символів.

Наступним кроком слід визначити кращий випадок представлення документів. Розглядаються всі категорії й обоє методу класифікації.

Таблиця 4.1 - Перша категорія. Без нормалізації

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
інше	0,432	0,613	0,507	0,556	0,484	0,517
доктори наук	0,302	0,813	0,441	0,600	0,188	0,286
кандидати наук	0,306	0,733	0,431	1,000	0,200	0,333
молоді вчені	0,708	0,447	0,548	0,543	0,658	0,595
молоді н.	0,257	0,450	0,327	0,300	0,900	0,450
молоді к.н	0,387	0,480	0,429	0,279	0,680	0,395
аспіранти	0,788	0,667	0,722	0,566	0,769	0,652
студенти	0,792	0,613	0,691	0,571	0,645	0,606
Середнє по категорії	0,497	0,602	0,51	0,552	0,566	0,479

Таблиця 4.2 - Перша категорія. Усі леми

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
Інше	0,538	0,677	0,600	0,533	0,516	0,525
доктори наук	0,326	0,938	0,484	0,333	0,188	0,240
кандидати наук	0,302	0,867	0,448	0,800	0,267	0,400
молоді вчені	0,781	0,658	0,714	0,600	0,711	0,651
молоді н.	0,320	0,400	0,356	0,297	0,950	0,452
молоді к.н	0,406	0,520	0,456	0,305	0,720	0,429
аспіранти	0,707	0,744	0,725	0,579	0,846	0,688
студенти	0,759	0,710	0,733	0,559	0,613	0,585
Середнє по категорії	0,517	0,689	0,565	0,5	0,601	0,496

Таблиця 4.3 - Перша категорія. Підм. & прис. & дієслова

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
інше	0,395	0,548	0,459	0,471	0,516	0,492
доктори наук	0,359	0,875	0,509	0,417	0,313	0,357
кандидати наук	0,294	1,000	0,455	0,571	0,267	0,364
молоді вчені	0,786	0,579	0,667	0,574	0,711	0,635
молоді н.	0,290	0,450	0,353	0,300	0,900	0,450
молоді к.н	0,452	0,560	0,500	0,300	0,720	0,424
аспіранти	0,763	0,744	0,753	0,564	0,795	0,660
студенти	0,786	0,710	0,746	0,571	0,645	0,606
Середнє по категорії	0,516	0,683	0,555	0,47	0,608	0,499

Таблиця 4.4 - Перша категорія. Підм. & прис.

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
інше	0,467	0,677	0,553	0,516	0,516	0,516
доктори наук	0,308	0,750	0,436	0,385	0,313	0,345
кандидати наук	0,298	0,933	0,452	0,462	0,400	0,429
молоді вчені	0,774	0,632	0,696	0,571	0,737	0,644
молоді н.	0,314	0,550	0,400	0,300	0,900	0,450
молоді к.н	0,421	0,640	0,508	0,322	0,760	0,452
аспіранти	0,750	0,769	0,759	0,681	0,821	0,744
студенти	0,767	0,742	0,754	0,545	0,581	0,563
Середнє по категорії	0,512	0,712	0,57	0,473	0,629	0,518

Отримані значення F-score для розглянутих вище випадків зведемо в результуючі таблиці

Аналіз таблиці показує, що для ряду класів вдається досягтися прийнятних результатів, для окремих класів результати отримані досить погані. Зокрема, із класів для яких отримані прийнятні результати виділимо наступні: студенти, аспіранти, молоді вчені. Це зв'язане в тому числі з тим, що в оголошеннях, що ставляться до цих класів, найчастіше в явному виді вказуються й студенти, і аспіранти й молоді вчені (або вказується, що вчені молодше певного віку), причому в половині випадків оголошення одночасно орієнтовані й на аспірантів, і на молоді вчені. Гірше всього є справи із класифікацією для класів: кандидати наук, молоді кандидати наук, молоді доктори наук. У першу чергу це можна зв'язати з тим, що в оголошеннях по цих класах, часто лише побічно (без явної вказівки) можна визначити, що оголошення ставиться до цих

класів учасників. До того ж набір таких оголошень і типи оголошень більш різноманітний, чому типи оголошень для студентів і аспірантів.

Таблиця 4.5 - Результуюча таблиця по першій категорії

			НБК			С 4.5		
клас	Ненорм	усі	с+п+г	с+п	ненорм	усі	с+п+г	с+п
інше	0,517	0,525	0,492	0,516	0,507	0,6	0,459	0,553
доктори наук	0,286	0,24	0,357	0,345	0,441	0,484	0,509	0,436
кандидати наук	0,333	0,4	0,364	0,429	0,431	0,448	0,455	0,452
молоді вчені	0,595	0,651	0,635	0,644	0,548	0,714	0,667	0,696
молоді н.	0,45	0,452	0,45	0,45	0,327	0,356	0,353	0,4
молоді к.н	0,395	0,429	0,424	0,452	0,429	0,456	0,5	0,508
аспіранти	0,652	0,688	0,66	0,744	0,722	0,725	0,753	0,759
студенти	0,606	0,585	0,606	0,563	0,691	0,733	0,746	0,754
Зважене середнє по категорії	0,479	0,496	0,499	0,518	0,51	0,565	0,555	0,57

Аналізуючи в цілому роботу класифікаторів відзначимо наступне:

Практично у всіх випадках (крім молодих докторів наук) алгоритм С4.5 показує більш високі результати, для випадків, коли слова були наведені до лем. Також найбільш гарні результати отримані у випадку представлення текстів за допомогою лем іменників і прикметників документа. Виключення становлять клас молоді вчені, коли переважніше представляти текст за допомогою всіх лем

документа, а також класи кандидатів і докторів наук, коли крім іменників і прикметників корисним є використання дієслів.

Проаналізуємо роботу класифікаторів, для випадків, коли в термінах F-score були отримані не високі результати. У першу чергу це молоді доктори наук і кандидати. Для обох класів, для випадку, коли отримані максимальні значення F-score ми спостерігаємо високу повноту (1,000 для кандидатів наук і 0,950 для молодих докторів наук) і низьку точність. Це говорить про те, що серед отриманих оголошень у цих класах будуть майже всі потрібні (тобто ми практично не втратимо оголошень про конкурси в цих класах), але при цьому в ці класи потрапить багато зайвих оголошень. Проте, з погляду практики високе значення повноти нам видасться більш важливим, чому точності. Останнє згладжує невисокі результати, отримані з погляду оцінки F-score.

Без нормалізації.

Таблиця 4.6 - Тип конкурсу. Без нормалізації

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
інше	0,263	0,652	0,375	0,300	0,130	0,182
проекти & гранти	0,583	0,341	0,431	0,500	0,024	0,047
премії, стипендії & виконані роботи	0,815	0,595	0,688	0,900	0,243	0,383
наукова мобільність	0,800	0,414	0,545	0,292	0,966	0,448
Стартапи & інноваційні проекти	0,545	0,800	0,649	0,519	0,933	0,667
Зважене середнє по категорії	0,631	0,517	0,533	0,531	0,379	0,298

Таблиця 4.7 - Тип конкурсу. С4.5. Матриця неточностей

Номера класів	1	2	3	4	5
1	15	4	1	0	3
2	22	14	2	0	3
3	8	3	22	3	1
4	11	2	1	12	3
5	1	1	1	0	12

Таблиця 4.8 - Тип конкурсу.Матриця неточностей

Номера класів	1	2	3	4	5
1	3	1	0	15	4
2	7	1	0	26	7
3	0	0	9	26	2
4	0	0	1	28	0
5	0	0	0	1	14

Проведена лематизація, залишені всі слова

Таблиця 4.9 - Тип конкурсу. Усі леми

	С4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
інше	0,395	0,652	0,492	0,429	0,130	0,200
проекти & гранти	0,792	0,463	0,585	0,667	0,049	0,091
премії, стипендії & виконані роботи	0,793	0,622	0,697	0,929	0,351	0,510
наукова мобільність	0,759	0,759	0,759	0,315	1,000	0,479
Стартапи & інноваційні проекти	0,440	0,733	0,550	0,448	0,867	0,591
Зважене середнє по категорії	0,686	0,621	0,630	0,603	0,414	0,345

Таблиця 4.10 - Тип конкурсу. C4.5. Матриця неточностей

Номера класів	1	2	3	4	5
1	15	3	0	0	5
2	9	19	3	4	6
3	8	0	23	3	3
4	4	1	2	22	0
5	2	1	1	0	11

Таблиця 4.11 - Тип конкурсу. Матриця неточностей

Номера класів	1	2	3	4	5
1	3	1	0	14	5
2	4	2	1	25	9
3	0	0	13	22	2
4	0	0	0	29	0
5	0	0	0	2	13

Проведена лематизація, залишені іменники, прикметники й дієслова.

Таблиця 4.12 - Тип конкурсу. Підм. & прис. & дієслова

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
інше	0,356	0,696	0,471	0,429	0,130	0,200
проекти & гранти	0,750	0,293	0,421	0,667	0,049	0,091
премії, стипендії & виконані роботи	0,741	0,541	0,625	0,929	0,351	0,510
наукова мобільність	0,568	0,724	0,636	0,330	1,000	0,496
Стартапи & інноваційні проекти	0,400	0,533	0,457	0,455	1,000	0,625
Зважене середнє по категорії	0,612	0,531	0,528	0,606	0,428	0,351

Таблиця 4.13 - Тип конкурсу. C4.5. Матриця неточностей

Номера класів	1	2	3	4	5
1	16	2	0	0	5
2	13	12	3	9	4
3	8	1	20	5	3
4	5	0	3	21	0
5	3	1	1	2	8

Таблиця 4.14 - Тип конкурсу. Матриця неточностей

Номера класів	1	2	3	4	5
1	3	1	0	13	6
2	4	2	1	24	10
3	0	0	13	22	2
4	0	0	0	29	0
5	0	0	0	0	15

Проведена лематизація, залишені іменники й прикметники.

Таблиця 4.15 - Тип конкурсу. Підм. & прис.

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
інше	0,356	0,696	0,471	0,429	0,130	0,200
проекти & гранти	0,750	0,293	0,421	0,667	0,049	0,091
премії, стипендії & виконані роботи	0,750	0,568	0,646	0,938	0,405	0,566
наукова мобільність	0,656	0,724	0,689	0,354	1,000	0,523
Стартапи & інноваційні проекти	0,458	0,733	0,564	0,405	1,000	0,577
Зважене середнє по категорії	0,639	0,559	0,555	0,608	0,441	0,366

Таблиця 4.16 - Тип конкурсу. C4.5. Матриця неточностей

Номера класів	1	2	3	4	5
1	16	2	0	0	5
2	17	12	3	4	5
3	6	1	21	7	2
4	4	0	3	21	1
5	2	1	1	0	11

Таблиця 4.17 - Тип конкурсу. НБК. Матриця неточностей

Номера класів	1	2	3	4	5
1	3	1	0	13	6
2	4	2	1	21	13
3	0	0	15	19	3
4	0	0	0	29	0
5	0	0	0	0	15

Приведемо результуючу таблицю значень F-score.

Аналіз отриманих результатів показує, що алгоритм C4.5 більше підходить для розв'язку завдання класифікації для даної категорії, причому кращі результати досягаються, якщо всі слова тексту були наведені до нормальної форми й не використовувався відбір термінів вроздріб мови.

Виключення становить клас «Стартапи & інноваційні проекти» для якого кращі результати показав

Наївний байесівський класифікатор, причому для випадку, коли слова в тексті не приводилися до нормальної форми й не було відбору термінів вроздріб мови.

Таблиця 4.18 - Результуюча таблиця по другій категорії

			НБК			с 4.5		
Клас	ненорм	усі	с+п+г	с+п	ненорм	усі	с+п+г	с+п
Інше	0,182	0,2	0,2	0,2	0,375	0,492	0,471	0,471
проекти & гранти	0,047	0,091	0,091	0,091	0,431	0,585	0,421	0,421
премії, стипендії & виконані роботи	0,383	0,51	0,51	0,566	0,688	0,697	0,625	0,646
наукова мобільність	0,448	0,479	0,496	0,523	0,545	0,759	0,636	0,689
Стартапи & інноваційні проекти	0,667	0,591	0,625	0,577	0,649	0,55	0,457	0,564
Зважене середнє по категорії	0,298	0,345	0,351	0,366	0,533	0,63	0,528	0,555

Без нормалізації

Таблиця 4.19 - Тип оголошення. С4.5. Матриця неточностей

Номера класів	1	2	3	4
1	94	8	4	5
2	6	10	0	0
3	3	3	4	6
4	5	2	1	7

Таблиця 4.20 - Тип оголошення. Без нормалізації

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
оголошення про конкурс	0,870	0,847	0,858	0,703	1,000	0,825
оголошення результатів	0,435	0,625	0,513	0,000	0,000	0,000
загальна інформація	0,444	0,250	0,320	0,000	0,000	0,000
інформація для учасників	0,389	0,467	0,424	0,000	0,000	0,000
Зважене середнє по категорії	0,737	0,728	0,728	0,494	0,703	0,580

Таблиця 4.21 - Тип оголошення. НБК. Матриця неточностей

Номера класів	1	2	3	4
1	111	0	0	0
2	16	0	0	0
3	16	0	0	0
4	15	0	0	0

Проведена лематизація, залишені всі слова.

Таблиця 4.22 - Тип оголошення. C4.5. Матриця неточностей

Номера класів	1	2	3	4
1	94	6	7	4
2	8	8	0	0
3	8	1	6	1
4	3	1	4	7

Таблиця 4.23 - Тип оголошення. Усі леми

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
оголошення про конкурс	0,832	0,847	0,839	0,703	1,000	0,825
оголошення результатів	0,500	0,500	0,500	0,000	0,000	0,000
загальна інформація	0,353	0,375	0,364	0,000	0,000	0,000
інформація для учасників	0,583	0,467	0,519	0,000	0,000	0,000
Зважене середнє по категорії	0,726	0,728	0,726	0,494	0,703	0,580

Таблиця 4.24 - Тип оголошення. НБК. Матриця неточностей

Номера класів	1	2	3	4
1	111	0	0	0
2	16	0	0	0
3	16	0	0	0
4	15	0	0	0

Проведена лематизація, залишені іменники, прикметники й дієслова.

Таблиця 4.25 - Тип оголошення. C4.5. Матриця неточностей

Номера класів	1	2	3	4
1	89	6	8	8
2	7	9	0	0
3	7	3	4	2
4	6	3	0	6

Таблиця 4.26 - Тип оголошення. Підм. & прис. & дієслова

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
оголошення про конкурс	0,817	0,802	0,809	0,703	1,000	0,825
оголошення результатів	0,429	0,563	0,486	0	0,000	0,000
загальна інформація	0,333	0,250	0,286	0	0,000	0,000
інформація для учасників	0,375	0,400	0,387	0	0,000	0,000
Зважене середнє по категорії	0,489	0,503	0,492	0,494	0,703	0,580

Таблиця 4.27 - Тип оголошення. НБК. Матриця неточностей

Номера класів	1	2	3	4
1	111	0	0	0
2	16	0	0	0
3	16	0	0	0
4	15	0	0	0

Проведена лематизація, залишені іменники й прикметники.

Таблиця 4.28 - Тип оголошення. C4.5. Матриця неточностей

Номера класів	1	2	3	4
1	88	10	2	11
2	3	12	1	0
3	10	2	1	3
4	4	2	0	9

Таблиця 4.29 - Тип оголошення. Підм. & прис.

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
оголошення про конкурс	0,838	0,793	0,815	0,707	1,000	0,828
оголошення результатів	0,462	0,750	0,571	0,000	0,000	0,000
загальна інформація	0,250	0,063	0,100	0,000	0,000	0,000
інформація для учасників	0,391	0,600	0,474	1,000	0,067	0,125
Зважене середнє по категорії	0,698	0,696	0,685	0,592	0,709	0,594

Таблиця 4.30 - Тип оголошення. НБК. Матриця неточностей

Номера класів	1	2	3	4
1	111	0	0	0
2	16	0	0	0
3	16	0	0	0
4	14	0	0	1

Аналіз отриманих результатів показує, що алгоритм C4.5 краще підходить для розв'язку завдання класифікації для даної категорії, причому кращі результати досягаються, якщо всі слова тексту були наведені до нормальної форми й без використання відбору термінів вроздріб мови.

Виключення становить клас «оголошення про конкурс» для якого більш гарні результати отримані при виставі тексту за допомогою його слів без нормалізації. Також виключення становить клас «оголошення результатів» для якого кращий результат показав наївний байесівський класифікатор для випадку,

коли слова в тексті приводилися до нормальної форми й використовувалися тільки іменники й прикметники.

Приведемо результуючу таблицю.

Таблиця 4.31 - Результуюча таблиця по третій категорії

			НБК			С4.5		
Клас	ненорм	усі	с+п+г	с+п	ненорм	усі	с+п+г	с+п
оголошення про конкурс	0,825	0,825	0,825	0,2	0,858	0,839	0,809	0,815
оголошення результатів	0	0	0	0,828	0,513	0,5	0,486	0,571
загальна інформація	0	0	0	0	0,32	0,364	0,286	0,1
інформація для учасників	0	0	0	0	0,424	0,519	0,387	0,474
Зважене середнє по категорії	0,58	0,58	0,58	0,125	0,728	0,726	0,492	0,685

Без нормалізації.

Таблиця 4.32 - Масштаб конкурсу. С4.5. Матриця неточностей

Номера класів	1	2	3	4	5
1	7	9	4	0	0
2	3	30	8	0	0
3	11	8	19	2	3
4	1	0	1	13	0
5	0	0	0	1	14

Таблиця 4.33 - Масштаб конкурсу. Без нормалізації

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
не зазначене	0,318	0,350	0,333	0,667	0,200	0,308
Міжнародний	0,638	0,732	0,682	0,521	0,927	0,667
Місцевий	0,594	0,442	0,507	0,667	0,698	0,682
внутрішнвузівський	0,813	0,867	0,839	1,000	0,533	0,696
міський & регіональний	0,824	0,933	0,875	1,000	0,133	0,235
Зважене середнє по категорії	0,616	0,619	0,613	0,697	0,612	0,573

Таблиця 4.34 - Масштаб конкурсу. НБК. Матриця неточностей

Номера класів	1	2	3	4	5
1	4	12	4	0	0
2	0	38	3	0	0
3	2	11	30	0	0
4	0	5	2	8	0
5	0	7	6	0	2

Проведена лематизація, залишені всі слова.

Таблиця 4.35 - Масштаб конкурсу. C4.5. Матриця неточностей

Номера класів	1	2	3	4	5
1	9	5	5	1	0
2	8	30	3	0	0
3	10	10	19	2	2
4	0	0	0	14	1
5	0	1	0	1	13

Таблиця 4.36 - Масштаб конкурсу. Усі леми

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
не зазначене	0,333	0,450	0,383	0,667	0,200	0,308
Міжнародний	0,652	0,732	0,690	0,638	0,902	0,747
Місцевий	0,704	0,442	0,543	0,567	0,791	0,660
внутрішнвузівський	0,778	0,933	0,848	1,000	0,400	0,571
міський & регіональний	0,813	0,867	0,839	1,000	0,267	0,421
Зважене середнє по категорії	0,653	0,634	0,631	0,700	0,634	0,598

Таблиця 4.37 - Масштаб конкурсу. НБК. Матриця неточностей

Номера класів	1	2	3	4	5
1	4	7	9	0	0
2	0	37	4	0	0
3	2	7	34	0	0
4	0	6	3	6	0
5	0	1	10	0	4

Проведена лематизація, залишені іменники, прикметники й дієслова.

Таблиця 4.38 - Масштаб конкурсу. C4.5. Матриця неточностей

Номера класів	1	2	3	4	5
1	10	6	3	1	0
2	7	30	4	0	0
3	11	6	22	2	2
4	0	0	0	14	1
5	0	0	0	1	14

Таблиця 4.39 - Масштаб конкурсу. Підм. & прис. & дієслова.

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
не зазначене	0,357	0,500	0,417	0,667	0,200	0,308
Міжнародний	0,714	0,732	0,723	0,607	0,902	0,725
Місцевий	0,759	0,512	0,611	0,579	0,767	0,660
внутрішнвузівський	0,778	0,933	0,848	1,000	0,467	0,636
міський & регіональний	0,824	0,933	0,875	1,000	0,200	0,333
Зважене середнє по категорії	0,695	0,672	0,672	0,695	0,627	0,588

Таблиця 4.40 - Масштаб конкурсу. НБК. Матриця неточностей

Номера класів	1	2	3	4	5
1	4	8	8	0	0
2	0	37	4	0	0
3	2	8	33	0	0
4	0	6	2	7	0
5	0	2	10	0	3

Проведена лематизація, залишені іменники й прикметники.

Таблиця 4.42 - Масштаб конкурсу. C4.5. Матриця неточностей

Номера класів	1	2	3	4	5
1	12	5	2	1	0
2	7	30	4	0	0
3	12	7	20	2	2
4	0	0	0	15	0
5	0	0	0	1	14

Таблиця 4.43 - Масштаб конкурсу. Підм. & прис.

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
не зазначене	0,387	0,600	0,471	0,667	0,200	0,308
міжнародний	0,714	0,732	0,723	0,621	0,878	0,727
місцевий	0,769	0,465	0,580	0,596	0,791	0,680
внутрішнвузівський	0,789	1,000	0,882	1,000	0,533	0,696
міський & регіональний	0,875	0,933	0,903	1,000	0,333	0,500
Зважене середнє по категорії	0,709	0,679	0,677	0,705	0,649	0,620

Таблиця 4.44 - Масштаб конкурсу. НБК. Матриця неточностей

Номера класів	1	2	3	4	5
1	4	8	8	0	0
2	0	36	5	0	0
3	2	7	34	0	0
4	0	5	2	8	0
5	0	2	8	0	5

Приведемо результуючу таблицю. Аналіз отриманих результатів показує, що більш удалим є використання алгоритму C4.5. У тих випадку, коли за допомогою цього алгоритму отримані результати вище чим в Наївного байесовского класифікатора вони значно вище, а от у зворотній ситуації результати C4.5 не сильно уступають кращим результатам НБК. У випадку використання алгоритму C4.5 оптимальним є представлення тексту за допомогою лемматизированных іменників і прикметників.

Таблиця 4.45 - Результуюча таблиця по четвертій категорії

			НБК			З 4.5		
Клас	ненорм	усі	с+п+г	с+п	ненорм	усі	с+п+г	с+п
не зазначене	0,308	0,308	0,308	0,308	0,333	0,383	0,417	0,471
міжнародний	0,667	0,747	0,725	0,727	0,682	0,69	0,723	0,723
місцевий	0,682	0,66	0,66	0,68	0,507	0,543	0,611	0,58
внутрішнвузівський	0,696	0,571	0,636	0,696	0,839	0,848	0,848	0,882
міський & регіональний	0,235	0,421	0,333	0,5	0,875	0,839	0,875	0,903
Зважене середнє по категорії	0,573	0,598	0,588	0,62	0,613	0,631	0,672	0,677

За підсумками експериментів, було виявлено, робота якого класифікатора й при якому способі представлення документів отримані найкращі результати для кожної категорії окремо.

Висновки по розділу 4

У ході даної роботи розроблявся інструмент для автоматичної класифікації текстових документів, що містять інформацію з наукової сфери. Вирішувалися такі завдання, як: розробка навчальної й тестової множин, вибір моделі представлення документа, аналіз можливостей інформаційної системи, вивчення двох алгоритмів машинного навчання – дерева побудови розв'язків і наївного байесівський методу.

Розглянуті різні підходи, що впливають на якість класифікації. За результатами проведеного дослідження для кожної категорії даних були визначені параметри, при яких були отримані найкращі результати.

Загальні висновки

Порівняння класичних алгоритмів показало, що метод визначення кластерів на множинах, отриманих з матриці подібності, показав себе малопридатним для розв'язку поставленого завдання, тому що має тенденцію до утворення великої кількості дуже дрібних груп. Алгоритм Роккіо показав трохи кращі результати: оскільки в даному методі класифікації відбувається навколо вибіркових документів, то стало можливим поява достатньо великих класів. Однак обчислена щільність простору документів виявилася такою, що більша частина документів не ввійшла ні в один кластер.

Більш якісний результат показав жадібний алгоритм. Його використання привело до формування масиву, у якому кожний кластер містить у середньому порядку 6-10 записів. При цьому, незважаючи на необхідність побудови матриці подібності, тимчасові витрати несуттєво відрізнялися від необхідних в алгоритмі Роккіо. Таким чином, у порівнянні з алгоритмом Роккіо жадібний алгоритм має ряд переваг:

1. Відсутня проблема занадто великої кількості великих кластерів.
2. Відсутня проблема занадто великої кількості дрібних кластерів.
3. Неможлива поява документів, що не потрапили ні в один кластер.
4. Немає проблеми визначення профілів документів, тобто центрів, навколо яких формуються кластери.

Однак при класифікації великих баз таке збільшення складності стає не настільки істотним, до того ж для створення системи, що автоматизує процес відбору даних, дослідження бази даних потрібно проводити тільки один раз.

Перелік посилань

1. Hartley, R.V.L., Transmission of Information. // Bell Systems Technical Journal, 7 July 1928, pp 535-563

2. Hull, D.A.: Stemming Algorithms - A Case Study for Detailed Evaluation in Journal of the American Society for Information Science 47(1), 1986, pp 70-84,

3. Pantel P., Turney P. Kantrowitz, M: Vector Space Models of Semantics // Journal of Artificial Intelligence Research 37, 2010, pp 141-188

4. Derose, Steven J. Stochastic Methods for Resolution of Grammatical Category Ambiguity in Inflected and Uninflected Languages. 1990. P 566

5. Miyao Y. From Linguistic Theory to Stochasticanalysis: Corpus-oriented Grammar Development and Feature Forest Methodsphd thesis, University oforokyo. 2006.

6. Porter M.F. An algorithm for suffix stripping / M.F. Porter // Program. - 1980. - Volume 14, № 3. - P. 130-137.

7. Quinlan J. Resolution : proframs for machine learning. San Mateo, Calif. :Morgan Kaufmann Publishers, c1993. P. 302

8. Ceriel, J. Grune, D. Parsing Techniques. A Practical Guide, 2007 P. 662

9. Green G. M., Morgan J. L., Practical guide to Syntactic analysis. 2001. P 14

10. Golub G. van Loan C. Matrix computations. Johns Hopkins University Press; 3rd edition (October 15, 1996) P. 728

11. Michie D., Spiegelhalter D.J., Taylor C.C.. Machine Learning, Neural and Statistical Classification. February 17, 1994. P. 290

12. Rokach L., Maimon O. Data Mining with Decision Trees. 2007. P264

13. Salton G., Wong A., Yang C.S., From Frequency to Meaning for automatic indexing

14. Srivastava A., Sahami M.. Text Mining: Classification, Clustering, and Applications. 2009. P. 328.

15. G.H. John, P. Langley, Estimating continuous distributions in Bayesian classifiers, in: Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence, 1995, pp. 338-345

16. Kantrowitz, M. Stemming and its effects on TFIDF ranking / M. Kantrowitz, B. Mohit, V. Mittal // In Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval. - 2000. - NY, USA: ACM Press. - P . 357-359.

17. Singal A., Salton G., Mitra M., Buckley C. Document Lenght Normalization. Information Processing and Management. Technical Report TR95-1529, Department of Computer Science, Cornell University, Ithaca, New York, July 1995.

Додатки

Таблиця 1 - Результуюча таблиця по першій категорії

			НБК			С 4.5		
клас	Ненорм	усі	с+п+г	с+п	ненорм	усі	с+п+г	с+п
інше	0,517	0,525	0,492	0,516	0,507	0,6	0,459	0,553
доктори наук	0,286	0,24	0,357	0,345	0,441	0,484	0,509	0,436
кандидати наук	0,333	0,4	0,364	0,429	0,431	0,448	0,455	0,452
молоді вчені	0,595	0,651	0,635	0,644	0,548	0,714	0,667	0,696
молоді н.	0,45	0,452	0,45	0,45	0,327	0,356	0,353	0,4
молоді к.н	0,395	0,429	0,424	0,452	0,429	0,456	0,5	0,508
аспіранти	0,652	0,688	0,66	0,744	0,722	0,725	0,753	0,759
студенти	0,606	0,585	0,606	0,563	0,691	0,733	0,746	0,754
Зважене середнє по категорії	0,479	0,496	0,499	0,518	0,51	0,565	0,555	0,57

Таблиця 2 - Тип оголошення. Без нормалізації

	C4.5			НБК		
Клас	Precision	Recall	F-score	Precision	Recall	F-score
оголошення про конкурс	0,870	0,847	0,858	0,703	1,000	0,825
оголошення результатів	0,435	0,625	0,513	0,000	0,000	0,000
загальна інформація	0,444	0,250	0,320	0,000	0,000	0,000
інформація для учасників	0,389	0,467	0,424	0,000	0,000	0,000
Зважене середнє по категорії	0,737	0,728	0,728	0,494	0,703	0,580

УДК 004.4

Прокопов Р. І., Манзюк Е. А., Скрипник Т. К.

Хмельницький національний університет

ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ ВИЗНАЧЕННЯ ПОДІБНОСТІ ДОКУМЕНТІВ

В роботі досліджено методи кластеризації, що використовують у якості критерію для порівняння тільки заздалегідь заданих елементів бібліографічного опису документів, не враховуючи індивідуальні пошукові можливості даних документів і думки споживачів про їхню корисність.

The paper studies clustering methods that use as a criterion for comparing only predefined elements of the bibliographic description of documents, without taking into account the individual search capabilities of these documents and consumers' opinions about their usefulness.

Щорічно у світі публікуються мільйони наукових статей. Навіть у вузькоспеціалізованих галузях науки переглядати весь обсяг інформації практично неможливо.

Крім цього в кожного дослідника за роки його роботи утворюється картотека бібліографічних описів статей, книг і т.д., що представляють для нього інтерес. Основний критерій їх відбору - особисті інтереси вченого. У цей час такі картотеки зберігаються, як правило, на електронних носіях. Це дозволяє організовувати інтегровані картотеки шляхом об'єднання ресурсів спільно працюючих дослідників.

Таким чином, виникає завдання автоматизації процесу відбору публікацій з електронних баз даних, які можуть становити інтерес для конкретного дослідника або групи спільно працюючих дослідників. Для знаходження потрібної статті дослідник звертається або до реферативних журналів, або до їхніх електронних аналогів[1]. Тому що існують досить ефективні алгоритми пошуку конкретної публікації на електронних носіях, те найбільш актуальної серед проблем інформаційного пошуку на даний момент є завдання знаходження по даному документу класу схожих по змісту документів.

У даній роботі вирішується завдання автоматизації процесу відбору публікацій з даних. Для баз даних, які можуть становити інтерес для конкретного дослідника або групи спільно працюючих дослідників. Із цією метою на першому етапі роботи досліджуються алгоритми виміру міри подібності між двома документами електронної бази даних, а також алгоритми кластеризації документів, що становлять цю базу. У якості шкал для визначення міри пропонується брати атрибути бібліографічного опису документів.

Під кластирізацією розуміється процес розбивки множини документів електронної бази на класи, при якому елементи, поєднувані в один клас, мають більша подібність, ніж елементи, що належать різним класам (у нашому випадку порівняння документів при формуванні кластерів ведеться по атрибутах їх бібліографічного опису [2]. Кожний кластер при цьому звичайно описується за допомогою одного або декількох ідентифікаторів, названих профілем або центроїдом. Профіль кластера може бути представлений деяким формальним об'єктом, розташованим у центрі кластера, або будь-яким представницьким об'єктом, здатним характеризувати інші об'єкти цього кластера (докладніше про різні методи визначення центроїдів буде розказано нижче). З використанням поняття центроїда можна шукати схожі документи, порівнюючи пошукові запити спочатку із профілями кластерів[3], а потім, перевіряти записи, що входять у кластери, та мають дуже близькі профілі.

Завдання полягає у знаходженні подібності на множині документів.

Як уже згадувалося, у якості шкал для визначення міри подібності між двома документами використовуємо атрибути бібліографічного опису даних документів. Перелічимо основні елементи бібліографічного опису, що заносяться в картотеку документів: автори; заголовок; назва журналу або видавництва; рік виходу; тому, номер, сторінки (для публікацій у періодичних виданнях); анотація; коди класифікатора; ключові слова.

Викладемо алгоритм визначення міри подібності нового документа з документами з наявного набору (тобто особистої бібліографічної бази). Кількісна характеристика міри подібності визначається на множини документів D у такий спосіб:

$$m: D \times D \to [0, 1],$$

причому функція m у випадку повної подібності ухвалює значення 1, у випадку повної відмінності - 0. Обчислення міри подібності здійснюється по формулі виду

$$m\_(d\_1,d\_2) = \sum \llbracket a\_i \, m\_i (d\_1,d\_2) \rrbracket , \quad (1)$$

де i – номер елемента (атрибута) бібліографічного опису,

a, – вагові коефіцієнти, причому $\sum \llbracket a\_i = 1 \rrbracket$,

mi (d1,d2) – міра подібності по i-му елементу(іншими словами, по i-й шкалі).

Оскільки в описуваній ситуації практично всі шкали - номінальні, то міра подібності по i-й шкалі визначається в такий спосіб: якщо значення i- х атрибутів документів збігаються, то міра близькості рівний 1, інакше 0. При цьому необхідно враховувати, що значення атрибутів можуть бути складовими. У такому випадку mi = ni 1/ni 0, де ni0 = max {ni0(d1), ni0(d2)}, а ni0(dj) – загальна кількість елементів, що становлять значення i-го атрибута документа dj, ni1 – кількість співпадаючих елементів.

Помітимо, що викладений алгоритм виміру міри подібності, може бути покладений в основу деякої експертної системи, що володіє певними продуктивними правилами. Так, значення вагових коефіцієнтів ai у формулі (1) може визначатися передбачуваною апостеріорною вірогідністю даних відповідної шкали. Наприклад, повний (або навіть «майже повний») збіг значень атрибута

«автори» документів d1 і d2 більш вагомі у випадку, коли кількість значень цього атрибута в документі d1 досить велика (у порівнянні з випадком, коли документ d1 має всього одного автора). У такій ситуації можемо збільшувати значення відповідного вагового коефіцієнта у формулі (1) з одночасним пропорційним зменшенням інших коефіцієнтів.

Методи кластеризації документів. Основна проблема кластеризації документів полягає в такому рознесенні документів по групах, при якому елементи кожної групи були б настільки подібні один з одним, щоб у деяких випадках можна було зневажити їхніми індивідуальними особливостями. Зокрема, робити пошук у систематизованому файлі набагато легше, чим у несистематизованому, тому що групи документів, профілі яких не мають подібності з пошуковим приписанням, не включаються в поглиблений процес пошуку. При кластеризації документів важливо прийти до розумного компромісу щодо розміру кластерів, уникаючи як формування великого числа дуже дрібних кластерів (що знижує ефективність кластеризації як виділення множин схожих документів), так і невеликої кількості дуже великих класів (що може викликати зменшення точності пошуку).

Прийнято розрізняти ряд завдань класифікації: формування кластерів на основі відомостей (властивостей і характеристик) про об'єкти; віднесення об'єктів до сформованих кластерів або кластерів, що перебувають у процесі формування; витяг інформації, необхідної для ідентифікації й опису класів документів. Властиво формування класів виконується звичайно на основі зіставлення векторів документів, причому клас визначається як множина усіх об'єктів, що мають досить високі значення коефіцієнта подібності. Складання характеристик класу еквівалентно побудові профілю; віднесення об'єктів до класів залежить від ступеня подібності між ідентифікаторами об'єктів і профілями класів.

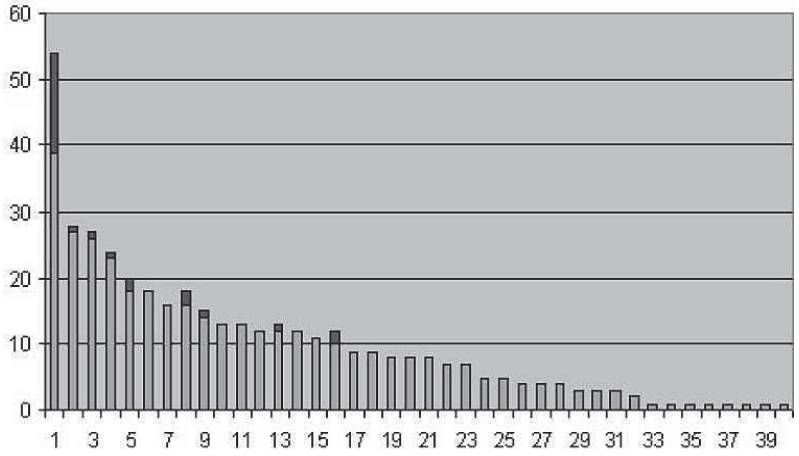


Рис. 1 Алгоритм кластиризації Роккіо

Виявлення оптимального способу завдання міри подібності на множини документів

У якості потенційно придатних для розв'язку поставленого завдання були проаналізовано три класичні методи кластеризації документів: кластеризація шляхом знаходження клік у повній матриці подібності документів , кластеризація по методу Роккіо і метод, що базується на так званому жадібному алгоритмі, а також новий алгоритм, заснований на використанні функції конкурентної подібності. Коротко викладемо суть перерахованих алгоритмів.

Процес знаходження подібності заснований на побудові повної матриці подібності, за допомогою якої кожній парі документів (d1, d2) ставиться у відповідність коефіцієнт подібності S(d1, d2). Звичайно вибирається граничне значення T, і матриця подібності приводиться до бінарного виду шляхом заміни всіх коефіцієнтів подібності таких, що .S'(d1, d2) > T, одиницею, а всіх інших - нулем. Далі шукані класи визначаються як значення, які можуть бути отримані з бінарного ряду подібності.

В алгоритмі Роккіо побудова матриці подібності заміняється перевіркою щільності простору деяких документів. У якості можливих центрів кластерів виступають тільки ті документи, які за результатами обчислень виявилися розташованими в щільних зонах простору. Документ відносять до того класу, подоба із центроїдом який виявився найбільш високим.

При використанні жадібного алгоритму в матриці подібності знаходять рядок (або стовпець - матрицям симетрична), сума компонентів якого буде максимальною. Документ, відповідний до цього рядка, визначається центром першого кластера та включають у кластер усі документи, коефіцієнти подібності до яких більше або рівні якогось наперед заданого граничного значення. Далі викидають усі документи, що потрапили в кластер, викреслюючи з матриці відповідні рядки та стовпці, після чого процес повторяться кілька раз, поки всі документи не будуть кластеризовані.

У методі кластеризації з використанням функції конкурентної подібності при визначенні міри подібності між двома документами розглядається конкурентна ситуація: розв'язок про приналежність документа d до першого кластера ухвалюється не в тому випадку, коли відстань r1 до цього кластера «мало», а коли воно менше відстані r2 до конкуруючого кластера. Для обчислення міри конкурентної подібності, обмірюваної в абсолютній шкалі, використовується нормована величина F12 = (r2 – r1)/(r2 + r1), називана функцією конкурентної подібності або Fris-Функцією (від Function of Rival Similarity). Зрозуміло, на первісному етапі кластеризації кластерів доводиться працювати з деякою модифікацією (редукцією) Fris-Функції, що використовує віртуальний кластер-конкурент. Суть алгоритму полягає в тому, що з використанням скороченої функції в якості центроїдів вибираються центри локальних «згустків» розподілу документів, після чого формуються лінійно роздільні кластери.

Вибір оптимального алгоритму. У якості практичної мети застосування проаналізованих алгоритмів стояло завдання автоматизації процесу відбору публікацій з електронних баз даних, які можуть становити інтерес для конкретного дослідника або групи спільно працюючих дослідників.

Знаходження оптимального алгоритму кластеризації. У якості міри на просторі документів використовується визначена раніше конструкція, однак порівняння ведеться по одному єдиному атрибуту - кодам класифікатора. Оскільки збіг даних кодів для групи документів є об'єктивним критерієм збігу тематики даних документів, то такий міра можна вважати ідеальної.

Завдання міри на множини документів, яка після кластеризації бази дасть результат, близький до результату з використанням міри, визначеної в п. 1.

Порівняння трьох класичних алгоритмів показало, що метод визначення кластерів на множинах, отриманих з матриці подібності, показав себе малопридатним для розв'язку поставленого завдання, тому що має тенденцію до утворення великої кількості дуже дрібних груп. Алгоритм Роккіо показав трохи кращі результати: оскільки в даному методі кластеризація відбувається навколо вибіркових документів, то стало можливим поява достатньо великих класів. Однак обчислена щільність простору документів виявилася такою, що більша частина документів не ввійшла ні в один кластер.

Більш якісний результат показав жадібний алгоритм. Його використання привело до формування кластерного масиву, у якому кожний кластер містить у середньому порядку 6-10 записів. При цьому, незважаючи на необхідність побудови матриці подібності, тимчасові витрати несуттєво відрізнялися від необхідних в алгоритмі Роккіо. Таким чином, у порівнянні з алгоритмом Роккіо жадібний алгоритм має ряд переваг.

1. Відсутня проблема занадто великої кількості великих кластерів.
2. Відсутня проблема занадто великої кількості дрібних кластерів.
3. Неможлива поява документів, що не потрапили ні в один кластер.
4. Немає проблеми визначення профілів документів, тобто центрів, навколо яких формуються кластери.

Однак при кластеризації великих баз таке збільшення складності стає не настільки істотним, до того ж для створення системи, що автоматизує процес відбору наукових публікацій, кластеризацію бази даних потрібно проводити тільки один раз.

Перелік посилань

1. Zhou et al., 2003b D. Zhou, J. Weston, A. Gretton, O. Bousquet and B. Schlkopf. Ranking on data manifolds. In Proceedings of NIPS'2003.
2. Manifold-Ranking Based Topic-Focused Multi-Document Summarization" [Електронний ресурс] DUC 2003. - Режим доступу http://www.ijcai.org/papers07/Papers/IJCAI07-467.pdf
3. Barzilay R. Sentence Ordering in Multidocu-ment Summarization. [Електронний ресурс] Computer Science at Co-lumbia University, Web seit, 2007. – Режим доступу http://www.cs.columbia.edu /nlp/papers/2001/barzilay_al_01.pdf

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ДИПЛОМНА РОБОТА МАГІСТРА

Інформаційна система для визначення подібності документів

Розробив ст. гр. КНм-19-1:
Прокопов Р.І.

Хмельницький - 2020

Класифікація текстових даних по темам - це визначення приналежності текстових даних до якої-небудь теми. Найбільш часто зустрічаються завдання класифікації текстових даних - це визначення емоційного забарвлення тексту та класифікація текстових даних за темами. Класифікація за темами (багатокласова класифікація) часто використовується для фільтрації текстових даних, коли необхідно видалити записи, що відносяться до теми, які не представляють інтересів для аналізу. Існує кілька підходів до класифікації тексту. Перший - ручний - представляє собою визначення класу документа вручну.
Другий підхід - написання правил на основі регулярних виразів.
Третій підхід базується на машинному навчанні. При цьому підході залежність класу від тексту зразка визначається автоматично..

Метою є розробка інформаційної системи для визначення подібності документів.
Для досягнення поставленої мети визначені наступні завдання:
1. Дослідити існуючі методи класифікації документів;
2. Розробити інформаційну систему для визначення подібності документів;
3. Виконати алгоритмічну та програмну реалізацію подібності документів.

Об’єкт дослідження. Нові підходи до вирішення задачі визначення подібності документів на підставі їх класифікації.

Предмет дослідження. Моделі, методи, підходи та засоби для визначення подібності документів.

У даній роботі вирішується завдання автоматизації процесу класифікації текстових даних. Із цією метою на першому етапі роботи досліджуються алгоритми виміру міри подібності між двома документами бази даних, а також алгоритми класифікації документів, що становлять цю базу. У якості шкал для визначення міри пропонується брати атрибути документів.

Основна проблема класифікації документів полягає в такому рознесенні документів по групах, при якому елементи кожної групи були б настільки подібні один з одним, щоб у деяких випадках можна було зневажити їхніми індивідуальними особливостями. Зокрема, робити пошук у систематизованому файлі набагато легше, чим у несистематизованому, тому що групи документів, профілі яких не мають подібності з пошуковим приписанням, не включаються в поглиблений процес пошуку.

Одним з найпоширеніших способів оцінки ваги терма для матриць термін-документ є TF-IDF (частота терміна × зворотна частота документа) сімейство вагових функцій (Спарк Джонс, 1972).

$$TF\frac{n_d}{\sum_D n_i}$$

де t – яке-небудь слово в документі d;

D – множина усіх документів корпуса;

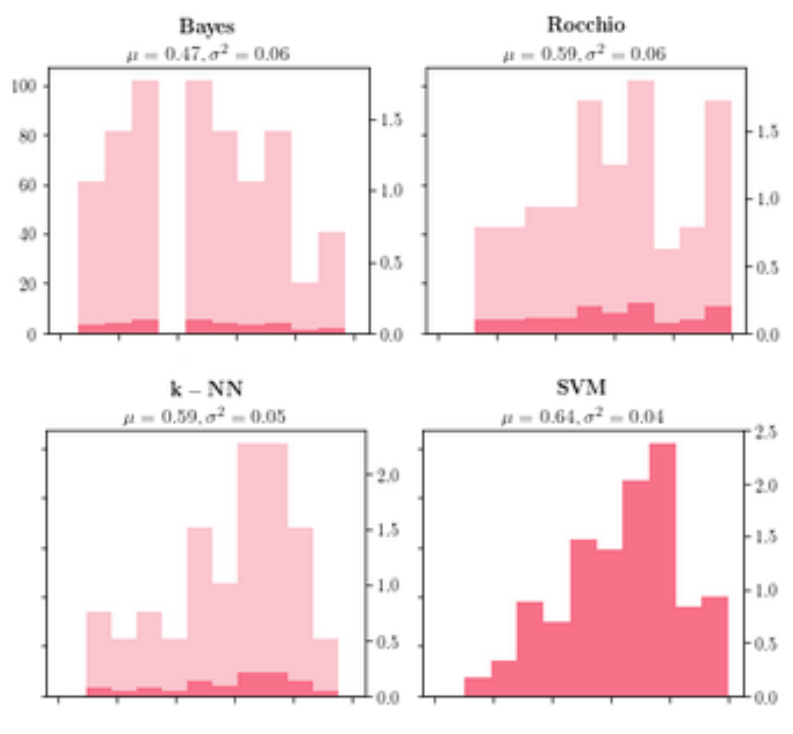
n_d і n_i – скільки раз слово з'явилося в розглянутому та і-*му* документах відповідно.

$$IDF(t, D) = \log \frac{|D|}{|(d_i \ni t)|}$$

де $|D|$ – число документів у колекції;
$|(d_i \ni t)|$ – скільки всього документів містять *t*.

Міра TF-IDF виходить перемножуванням двох співмножників:

$$TF - IDF\,(t, d, D) = TF(t, d) \times IDF(t, D)$$



Оцінка точності F1

Прозора гістограма нормалізована, непрозора обсолютна

Висновки

У результаті порівняння існуючих методів класифікації текстових даних можна отримати наступний висновок - найкращим підходящим методом для класифікації невеликого обсягу даних є метод опорних векторів. Метод Баєсовського класифікатора може бути використаний у якості альтернативного методу, за умови наявності додаткової інформації про розміщення даних. Алгоритм Роккіо показав трохи кращі результати: оскільки в даному методі класифікації відбувається навколо вибіркових документів, то стало можливим появи достатньо великих класів.

Дякую за увагу

Wed Dec 02 22:25:18 EET 2020, Петровський Сергій Степанович, Хмельницький національний університет, ХНУ

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 1.0%

Словники перевірки: en_US, ru_RU, ua_UA. **Помилок в документах: 6%**

	Документ		Сумарний збіг по Базі Даних	
ID: 82192 Назва: Інформаційна система для визначення подібності документів Додано в БД: 2020-12-02 Автора: Прокопов Роман Ігорович Керівники: Манзюк Е.А. Консультанти: Опоненти:	Символи	Лексеми	Символи	Лексеми
	64046	460	590 (1%)	10 (2%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

РІШЕННЯ КАФЕДРИ **КОМП'ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ** ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: **Інформаційна система для визначення подібності документів**
Автор: **Прокопов Р. І.**
Спеціальність: **122 Комп'ютерні науки**
Науковий керівник: **к.т.н. доцент Манзюк Е.А.**

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних). Робота приймається до захисту.	**відповідає**
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	-
3	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	-
4	Інше:	-

Підтвердження: Виявленні запозичення не є плагіатом так як є широко вживаними поняттями предметної області і складають 1.2%.

01.11.2020
Дата

Підпис керівника

Підпис завідувача кафедри

Хмельницький національний університет
Факультет програмування та комп'ютерних і телекомунікаційних систем
Кафедра комп'ютерних наук та інформаційних технологій

ВІДГУК ОПОНЕНТА

на дипломну роботу магістра

Магістра *гр. КНм-19-1 Прокопова Романа Ігоровича*

На тему: Інформаційна система для визначення подібності документів

1. Актуальність і значення теми

Найбільш актуальною серед проблем інформаційного пошуку на даний момент є завдання знаходження по даному документу класу схожих по змісту документів. Відповідно актуальність роботи визначається необхідністю розробки методів та засобів які дозволяють ефективно класифікувати текстову інформацію, що представлена у різного виду документах, як наукові статі так і публіцистичні огляди.

2.Оцінка якості та достовірності проведених досліджень.

Дослідження проводились із забезпеченням відповідності статистичним підходам та якісним оцінкам, які застосовні до подібних досліджень.

3. Оцінка запропонованих заходів та пропозицій, практичної цінності та ефективності.

Практична значимість дослідження полягає в тому, що описані методи та отримані результати можуть бути використані при розробці систем автоматичної класифікації документів та іншої текстової інформації.

4. Загальний висновок та оцінка

Робота належним чином та проведені дослідженні проведені в науково-практичному напрямку. Пояснювальна записка оформлена в відповідності із вимогами. Певні недоліки не знижують цінності роботи. За структурою та вирішеними задачами робота відповідає вимогам вищої школи та вимогам, що пред'являються до освітньо-кваліфікаційного рівня «магістр», автор заслуговує присвоєння кваліфікації магістра з комп'ютерних наук та інформаційних технологій.

Робота заслуговує на оцінку «задовільно».

Опонент Лисовецук Ж.К., к.т.н., доцент кафедри гідромеліоративного машинобудування та агроінженерії.