«АнтіСпам»
(шифр)

АНАЛІЗ І ВИЯВЛЕННЯ СПАМ ПОВІДОМЛЕНЬ У СОЦІАЛЬНИХ МЕРЕЖАХ

Галузь: Інформаційні системи і технології

2019/2020

АНОТАЦІЯ

У наші дні існує дуже багато різноманітних соціальних мереж та інших методів зв’язку. Також існує велика кількість непотрібної інформації, яка кожної секунди надходить до електронної пошти користувачів, тому зараз дуже актуальною є проблема боротьби зі спам повідомленнями.

Метою роботи є дослідження можливості застосування різних алгоритмів при розробці програмного забезпечення для виявлення спаму у текстовому контенті соціальних мереж та швидкісної фільтрації непотрібних повідомлень.

Поставлена мета передбачає вирішення таких завдань: а) аналіз особливостей виявлення спам повідомлень; б) аналіз існуючих методів боротьби зі спамом; в) аналіз особливостей перетворення тексту у вхідні данні алгоритмів; г) реалізація методів виявлення спаму на основі наївного байєсівського класифікатору, методу опорних векторів та багатошарової перцептронної нейромережі; д) проведення порівняльного аналізу результатів роботи використаних алгоритмів виявлення спаму.

Методи дослідження: теорія класифікації даних, ймовірнісні класифікатори, теорія нейронних мереж, статистичні методи обробки лінгвістичних даних.

Загальна характеристика роботи. Робота присвячена вирішенню науково-прикладної задачі виявлення спам повідомлень у текстовому контексті із використанням різних алгоритмів виявлення спаму. Було реалізовано та проведене дослідження 3-х алгоритмів: алгоритм з використанням наївного байєсівського класифікатору, методу опорних векторів та багатошарової перцептронної нейромережі.

Ключові слова: наївний байєсівський класифікатор, метод опорних векторів, векторизація тексту, багатошарова перцептронна нейромережа.

ЗМІСТ

ВСТУП

Об’єкт дослідження наданої роботи – спам повідомлення, предмет дослідження – дослідження специфіки виявлення спам повідомлень. Одна з головних проблем при модерації соціальних мереж або онлайн чатів – виявлення та видалення спам повідомлень. Не має значення чи це онлайн чат, чи електронна пошта, все одно необхідно виявляти повідомлення, яві не несуть ніякої корисної інформації. Видалення спам повідомлень є однією з найважливіших аспектів модерації будь якого засобу онлайн спілкування. Актуальність моєї дослідницької роботи полягає в тому, що в роботі проведений аналіз та порівняння методів розпізнавання спам повідомлень у текстовому контенту та розроблено програмне застосування перевірки повідомлення на наявність спаму.

Метою роботи є дослідження можливості використання різних алгоритмів при розробці програмного забезпечення для виявлення спаму у текстовому контенті соціальних мереж та швидкісного видалення непотрібних повідомлень.

Поставлена мета передбачає вирішення таких завдань:

– проведення аналізу особливостей та призначення виявлення спам повідомлень;

– аналіз деяких існуючих методів боротьби зі спам повідомленнями;

– обґрунтування використання алгоритмів виявлення спаму;

– розробка покрокової схеми роботи проекту програмного застосунку (ПЗ);

– порівняння різних алгоритмів виявлення спаму;

– створення програмної реалізації різних алгоритмів виявлення спам повідомлень.

Об’єкт дослідження – процес визначення спаму з використанням баєсового методу класифікації, методу опорних векторів та багатошарової перцептронної нейронної мережі.

Предмет дослідження – процес розпізнавання та блокування спаму у соціальній мережі з використанням баєсового методу класифікації, методу опорних векторів та багатошарової перцептронної нейронної мережі.

Методи досліджень – теорія нейронних мереж, теорія розпізнавання текстів, методи математичної статистики.

Наукова новизна – удосконалено метод розпізнавання та блокування спаму у соціальній мережі за рахунок комплексного застосування баєсового методу класифікації, методу опорних векторів та багатошарової перцептронної нейронної мережі.

РОЗДІЛ 1
АНАЛІЗ СПЕЦИФІКИ РОЗПІЗНАВАННЯ СПАМ-ПОВІДОМЛЕНЬ

1.1 Загальна характеристика спаму.

Спам –масове розсилання кореспонденції рекламного чи іншого характеру людям, які не висловили бажання її одержувати [1]. Передусім термін «спам» стосується рекламних електронних листів.

До різних видів спаму, як правило, відносять: рекламу; нігерійські листи; фішинг; інші види спаму.

Реклама.

Цей різновид спаму трапляється найчастіше. Деякі компанії рекламують свої товари чи послуги за допомогою спаму. Вони можуть розсилати його самостійно, але частіше замовляють це тим компаніям (чи особам), які на цьому спеціалізуються. Привабливість такої реклами в її порівняно низькій вартості і досить великому охопленню потенційних клієнтів.

Донедавна зовсім не було законів, які б забороняли чи обмежували таку діяльність. Тепер робляться спроби розробити такі закони, але це досить важко зробити. Складно визначити в законі, яка розсилка законна, а яка ні. Найгірше, що компанія (чи особа), що розсилає спам, може знаходитися в іншій країні. Для того, щоб такі закони були ефективними, необхідно погодити законодавство багатьох країн, що в найближчому майбутньому майже нереально. Проте в США, де такий закон уже прийнятий, є спроби притягнення спамерів до суду. Реклама незаконної продукції. За допомогою спаму часто рекламують продукцію, про яку не можна повідомити іншими способами, наприклад порнографію.

Нігерійські листи.

Іноді спам використовується для того, щоб виманити гроші в одержувача листа [2]. Найпоширеніший спосіб одержав назву «нігерійські листи», тому що дуже багато таких листів приходило з Нігерії. Такий лист містить повідомлення

про те, що одержувач листа може одержати якимось чином велику суму грошей, а відправник може йому в цьому допомогти. Потім відправник листа просить перерахувати йому трохи грошей: наприклад, для оформлення документів чи відкриття рахунку. Виманювання цієї суми і є метою шахраїв.

Фішинг.

Інший спосіб шахрайства за допомогою спаму одержав назву «фішинг» (англ. phishing від fishin – рибальство). В цьому разі спамер намагається виманити в одержувача листа номер його кредитних карток чи паролі доступу до електронних платіжних систем тощо. Такий лист, зазвичай, маскується під офіційне повідомлення від адміністрації банку. У ньому говориться, що одержувач повинен підтвердити відомості про себе, інакше його рахунок буде заблоковано, і наводиться адреса сайту, який належить спамерам, із формою, яку треба заповнити. Серед даних, що потрібно повідомити, є й ті, котрі потрібні шахраям.

Інші види спаму: розсилання листів релігійного змісту; масове розсилання для виведення поштової системи з ладу (доведення системи до відмови сервісу); масове розсилання від імені іншої особи, з метою викликати до неї негативне ставлення; масове розсилання листів, що містять комп'ютерні віруси (для їхнього початкового поширення).

Комп'ютерні віруси певного типу (поштові хробаки) поширюються за допомогою електронної пошти. Заразивши черговий комп'ютер, такий хробак шукає в ньому e-mail адреси й розсилає себе за цими адресами. Поштові хробаки часто підставляють випадкові адреси (зі знайдених на зараженому комп'ютері) у поле листа «Від кого». Недосконалі антивірусні програми на інших комп'ютерах відсилають на цю адресу повідомлення про знайдений вірус. У результаті десятки людей одержують повідомлення про те, що вони нібито розсилають віруси, хоча насправді вони не мають до цього жодного стосунку.

До основних способів поширення спаму насьогодні відносять [2]: електронну пошту; Usernet; мессенджери; підміну інтернет-трафіку; SMS-повідомлення; телефонні дзвінки тощо.

Електронна пошта.

Найбільший потік спаму поширюється через електронну пошту (e-mail). Станом на 2016 рік потік спаму в загальному трафіку електронної пошти становить близько 65 % за даними Cisco Systems.

Usernet.

Багато груп новин Usenet, особливо немодеровані, втратили користувачів і зараз містять переважно рекламу, часто навіть не за темою. Замість інших були створені модеровані конференції.

Мессенджери.

З появою великої кількості різноманітних месенджерів таких як Telegram, Viber та ін., спамери почали використовувати їх для своїх цілей.

Підміна інтернет-трафіку.

Відомі випадки коли деякі недобросовісні провайдери інтернет-зв'язку завдяки можливості підміни на льоту користувацького http-трафіку самовільно змінювали відображуваний користувачеві вміст отриманої ним з мережі веб-сторінки стороннього ресурсу або підмінювали її власною заради більшої реклами яких-небудь своїх додаткових послуг.

SMS-повідомлення.

Спам може поширюється не тільки через Інтернет. Найбільш поширеними є Нігерійські листи. Рекламні SMS-повідомлення, які надходять на особливо неприємні тим, що від них важче захиститися, і одержувач іноді повинен платити за кожне повідомлення. Це може бути досить велика сума, особливо якщо абонент використовує роумінг.

Телефонні дзвінки.

Деякі компанії можуть масово здійснювати велику кількість телефонних дзвінків з доволі агресивною рекламою завдяки поступовому перебору телефонних номерів, випадковій вибірці номера з певного діапазону або використанню готових баз даних, придбаних на чорному ринку чи завдяки доступу до телефонної інфраструктури – своїх власних (наприклад, мобільні оператори).

Захист від спаму.

Спам-повідомлення злочинцю практично нічого не коштують, але одержувач спаму зазвичай повинен оплачувати провайдерові час, використаний на одержання спаму. Також масове поширення спаму ускладнює роботу інформаційних систем та ресурсів, на них приходить дуже багата кількість непотрібного навантаження. Через масовість поштових розсилань користувач вимушений витрачати зайвий час на фільтрацію повідомлень, аби уникнути цього, користувачі використовуюсь протиспамові фільтри, які допомагають зберегти час. Але спам фільтри також можуть випадково стерти й важливе повідомлення, розпізнавши його як спам.

1.2 Способи боротьби зі спамом.

Найнадійніший засіб боротьби зі спамом – не дозволити спамерам роздобути вашу електронну адресу [3]. Це важке завдання, але деякі запобіжні заходи все ж варто вжити:

– не варто без необхідності публікувати адресу електронної пошти на веб-сайтах чи в групах Usenet;

– не потрібно реєструватися на підозрілих сайтах. Якщо якийсь корисний сайт вимагає реєстрації, можна вказати спеціально для цього створену адресу;

– ніколи не відповідати на спам і не переходити за посиланнями, які містяться в ньому. Цим ви тільки підтвердите, що користуєтеся своєю електронною адресою та будете отримувати ще більше спаму;

– вибираючи собі ім'я електронної пошти варто, за можливості, обирати довге й незручне для вгадування ім'я.

Існує програмне забезпечення (ПЗ) для автоматичного визначення спаму (так звані фільтри). Воно може застосовуватися кінцевими користувачами або на серверах. Це ПЗ має два основні підходи.

Перший полягає в аналізі змісту листа на основі чого робиться висновок, спам це чи ні. Якщо лист класифікований як спам, він може бути позначений, переміщений в іншу папку або навіть вилучений. Таке ПЗ може працювати як на сервері, так і на комп'ютері клієнта. При такому підході ви не бачите відфільтрованого спаму, але продовжуєте повністю платити витрати, пов'язані з його прийомом, тому що антиспамне ПЗ в будь-якому випадку одержує кожен спамерський лист (затрачаючи ваші гроші), а тільки потім вирішує, показувати його чи ні.

Другий підхід базується на класифікації відправника як спамера, не заглядаючи в текст листа. Для визначення застосовуються різні методи. Це ПЗ може працювати тільки на сервері, який безпосередньо приймає пошту. При такому підході можна зменшити витрати — гроші витрачаються тільки на спілкування зі спамерськими поштовими програмами (тобто на відмову приймати листи) і звертання до інших серверів (якщо такі потрібні) при перевірці. Виграш, однак, не такий великий, як можна було б очікувати. Якщо одержувач відмовляється прийняти лист, спамерська програма намагається обійти захист і відправити його іншим способом. Кожну таку спробу доводитися відбивати окремо, що збільшує навантаження на сервер.

Чорні списки.

У чорні списки заносяться IP-адреси комп'ютерів, про які відомо, що з них ведеться розсилання спаму [4]. Також широко використовуються списки комп'ютерів, які можна використовувати для розсилання — «відкриті релеї» і «відкриті проксі», а також – списки «діалапів» – клієнтських адрес, на яких не може бути поштових серверів. Можна використовувати локальний список або список який підтримує хтось інший. Завдяки простоті реалізації, широке

поширення одержали чорні списки, запит до яких здійснюється через службу DNS. Вони називаються DNSBL (DNS Black List). Цей метод вже не такий ефективний. Спамери знаходять нові комп'ютери для своїх цілей швидше, ніж їх встигають заносити в чорні списки. Крім того, кілька комп'ютерів, що відправляють спам, можуть скомпрометувати весь поштовий домен і тисячі законослухняних користувачів на невизначений час будуть позбавлені можливості відправляти пошту серверам, що використовують такий чорний список.

Авторизація поштових серверів.

Були запропоновані різні способи для підтвердження того, що комп'ютер, що відправляє лист, дійсно має на це право (Sender ID, SPF, Caller ID, Yahoo DomainKeys), але вони поки не розповсюджені.

На рис. 1.1 наведено принцип роботи чорних списків.

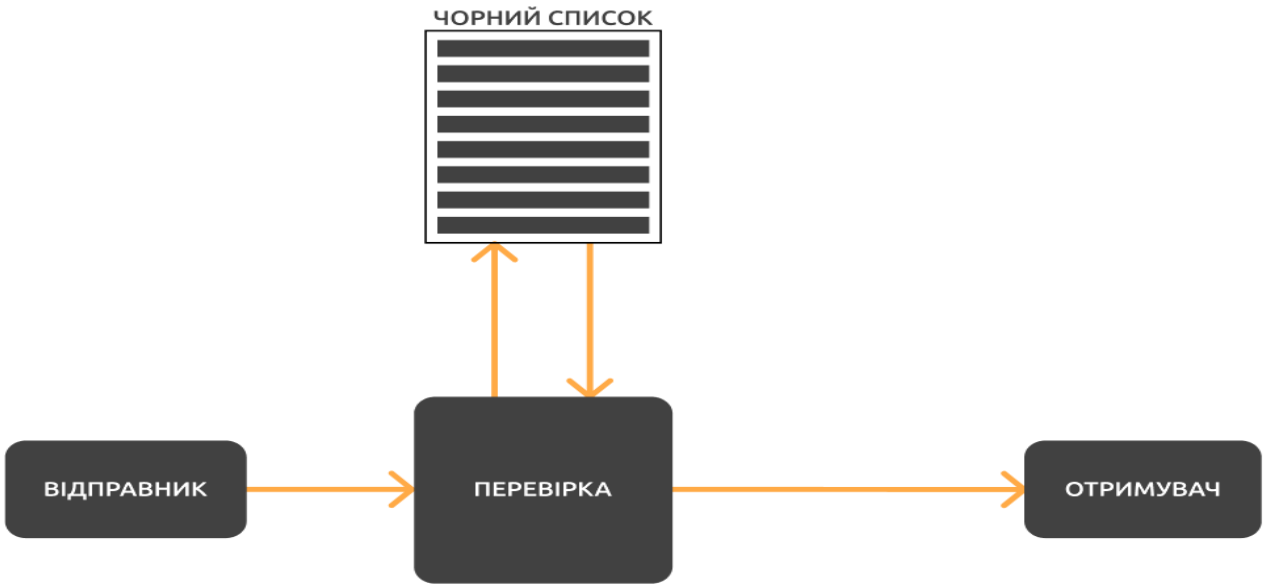


Рисунок 1.1 – Принцип роботи чорних списків

Сірі списки.

Метод сірих списків базується на тому, що «поведінка» програмного забезпечення, призначеного для розсилання спаму відрізняється від поведінки

звичайних поштових серверів, а саме, спамерські програми не намагаються повторно відправити лист при виникненні тимчасової помилки, як того вимагає протокол SMTP.

Спочатку всі невідомі сервери заносяться в «сірий список» і листи від них не приймаються. Серверові відправника повертається код тимчасової помилки, тому, звичайні листи (не спам) не втрачаються, а тільки затримується їхня доставка (вони залишаються в черзі на сервері відправника і доставляються при наступній спробі). Якщо сервер поводиться так, як очікувалося, він автоматично переноситься в білий список і наступні листи приймаються без затримки.

Цей метод в наш час дозволяє відсіяти до 90 % спаму, практично без ризику втратити важливі листи. Однак він теж не бездоганний.

Можуть помилково відсіватися листи з серверів, які не виконують рекомендації протоколу SMTP, наприклад, розсилки з сайтів новин.

Затримка при доставці листа може досягати півгодини (а іноді й більше), що неприйнятно для термінової кореспонденції. Цей недолік компенсується тим, що затримка вноситься тільки при відправці першого листа з раніше невідомої адреси.

Великі поштові служби використовують кілька серверів, з різними IP-адресами, більш того, можлива ситуація, коли кілька серверів по-черзі намагаються відправити той самий лист. Це може привести до дуже великих затримок при доставці листів.

Спамерські програми можуть удосконалюватися. Підтримка повторної посилки повідомлення реалізується досить легко і цілком нівелює даний вид захисту.

Статистичні методи фільтрації спаму.

Ці методи використовують статистичний аналіз змісту листа для прийняття рішення, чи є він спамом. Найбільшого успіху удалося досягти за допомогою алгоритмів, заснованих на теоремі Байеса. Для роботи цих методів потрібне «навчання» фільтрів, тобто потрібно використовувати розсортовані вручну листи

для виявлення статистичних особливостей нормальних листів і спаму. Після навчання на досить великій вибірці, вдається розпізнати до 95-97 % спаму.

Жорсткі вимоги до листів і відправників, наприклад – відмова прийому листів із заздалегідь неправильною зворотною адресою (листи з неіснуючих доменів), перевірка доменного імені за IP-адресою комп'ютера, з якого прийшов лист тощо. Дані заходи застаріли, відсівається тільки найпримітивніший спам – невелика кількість повідомлень. Але не нульова, тому застосування все ще має сенс.

У роботі розглядається статистичний баєсовий метод фільтрації спаму, застосування методу опорних векторів та багатошарової перцептронної нейромережі.

1.3 Використане програмне забезпечення.

Для створення програмного додатку були використані мова програмування Python та середовище програмування PyCharm.

Python – інтерпретована об'єктно-орієнтована мова програмування високого рівня зі строгою динамічною типізацією [5]. Розроблена в 1990 році Гвідо ван Россумом. Структури даних високого рівня разом із динамічною семантикою та динамічним зв'язуванням роблять її привабливою для швидкої розробки програм, а також як засіб поєднування наявних компонентів. Python підтримує модулі та пакети модулів, що сприяє модульності та повторному використанню коду. Інтерпретатор Python та стандартні бібліотеки доступні як у скомпільованій, так і у вихідній формі на всіх основних платформах.

PyCharm – інтегроване середовище розробки для мови програмування Python [6]. Надає засоби для аналізу коду, графічний зневаджувач, інструмент для запуску юніт-тестів і підтримує веб-розробку на Django. PyCharm розроблена компанією JetBrains на основі IntelliJ IDEA.

РОЗДІЛ 2
ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАСТОСУВАННЯ ВИЯВЛЕННЯ СПАМ ПОВІДОМЛЕНЬ

Проектування програмного застосування.

Створення даного програмного застосунку має базуватися на розробленій діаграмі.

У створюваному програмному застосуванні користувач повинен мати можливість вводити повідомлення для аналізу власноруч. Також користувач має можливість обрати алгоритм за яким він хоче щоб його текст було проаналізовано.

Користувач повинен мати можливість очистки робочого простору від використаної інформації.

Також, необхідно реалізувати збереження історії аналізу повідомлень до файлу, користувач повинен мати можливість вибрати каталог до якого буде здійснено збереження результатів.

У разі необхідності, користувач повинен мати можливість перегляду загальної інформації щодо створеного програмного продукту, зокрема, загальний опис програми, її призначення та інформації щодо роботи кожного з трьох алгоритмів аналізу повідомлень.

Діаграма варіантів використання програмного застосування наведена на рис. 2.1.

Послідовність застосування.

Користувач вводить у програму повідомлення(текст), який необхідно проаналізувати, далі користувач повинен вибрати алгоритм аналізу тексту. Програма реалізує обраний користувачем алгоритм та виводить відповідь на екран, також виводиться вірогідність з якою було розпізнано повідомлення.

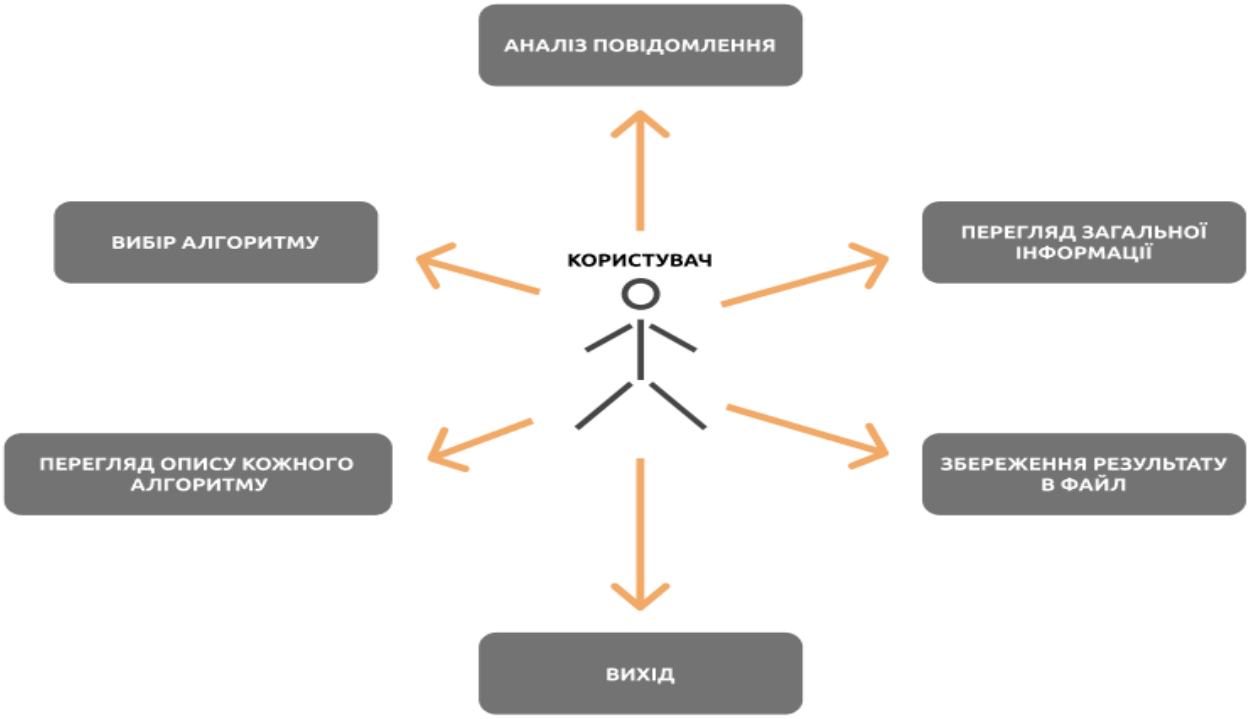


Рис. 2.1 – Діаграма інтерфейсу програмного продукту

Діаграма послідовності виконання програмного застосування наведена на рис 2.2.

Алгоритми аналізу спам повідомлень містять наступні етапи:

1) користувач вводить у програмне застосування початковий текст який має бути проаналізований;

2) програмне застосування розбивання початкового тексту на слова, потім кожне слово приводиться до початкового стану (інфінітиву), далі получений набір слів векторизується та передається на вхід до обраного алгоритму;

3) алгоритм аналізує отриманні дані та повертає результат у вигляді ймовірності приналежності отриманих даних до класу (у нашого алгоритму є два класи: спам та не спам);

4) отримані від алгоритму дані аналізуються та приводяться до зрозумілої користувачеві відповіді;

5) програмне застосування відображає результат аналізу отриманого повідомлення, також виводиться ймовірність, з якою було розпізнано повідомлення та назва алгоритму;

6) за бажанням користувача результат аналізу може бути записаний до обраного файлу

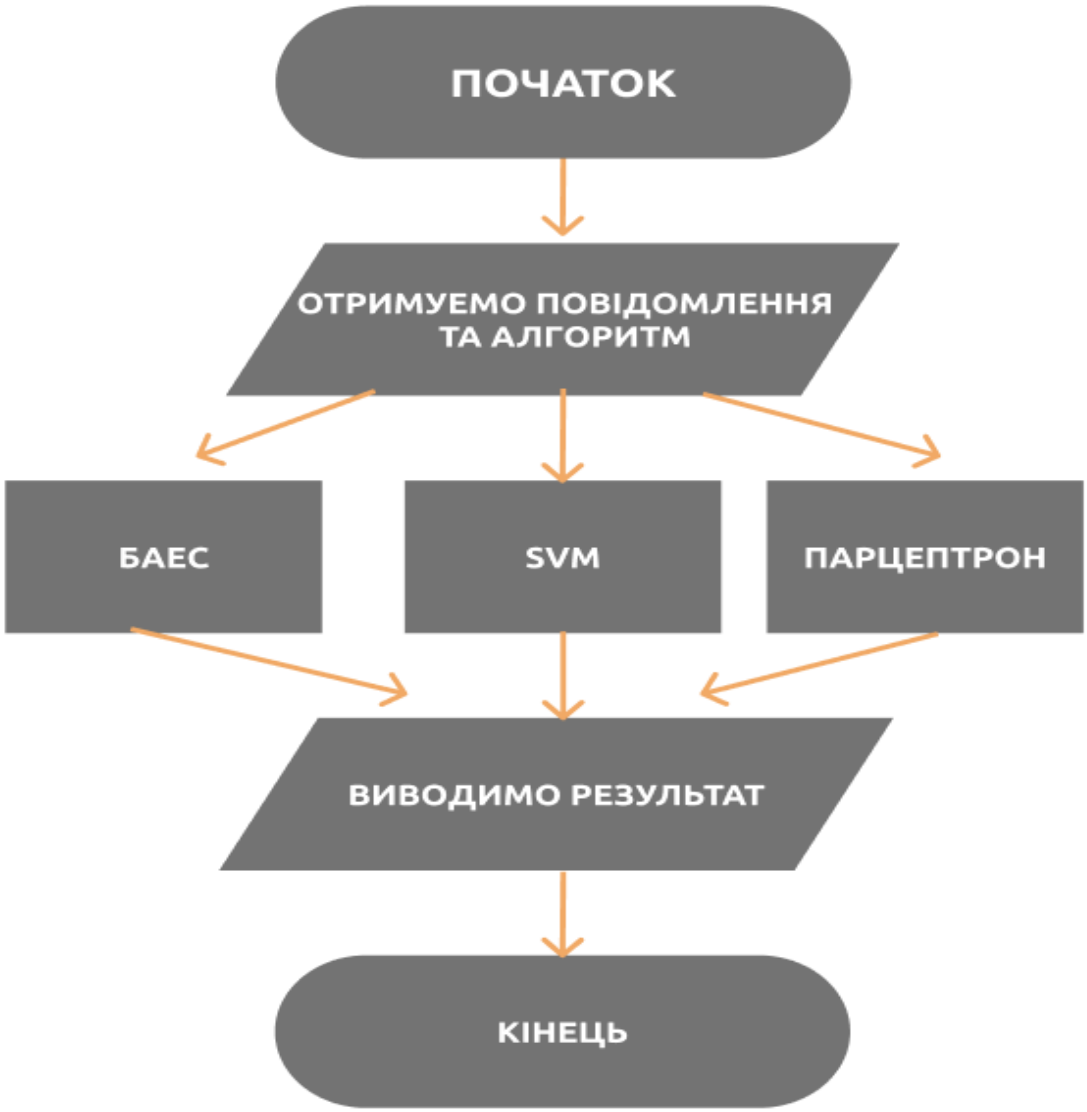


Рисунок 2.2 – Діаграма послідовності виконання програмного застосування

РОЗДІЛ 3
ОПИС ВИКОРИСТАНИХ АЛГОРИТМІВ

У даній роботі ми розглядаємо та порівнюємо три алгоритми розпізнавання спам повідомлень а саме: Наївний байесів класифікатор, Метод опорних векторів (SVM), Перцептрон.

Далі розберемо принцип роботи кожного з використаних алгоритмів.

3.1 Наївний баєсів класифікатор.

Наївний баєсів класифікатор – ймовірнісний класифікатор [15], що використовує теорему Баєса для визначення ймовірності приналежності спостереження (елемента вибірки) до одного з класів [15] при припущенні (наївному) незалежності змінних. Прикладом роботи цього метода може бути: розпізнавання спаму, аналіз емоційного забарвлення текстів [19], виявлення расизму у текстовій виборці, будь які системи обробки інформації [20] тощо.

Тобто, якщо на основі значень змінних можна однозначно визначити, до якого класу належить спостереження, байєсів класифікатор повідомить ймовірність приналежності до цього класу [16].

У проміжних же випадках, коли спостереження може з різною ймовірністю належати до різних класів, результатом роботи класифікатора буде вектор, компоненти якого є ймовірностями приналежності до того чи іншого класу.

Можна бачити, що ідеальний байєсів класифікатор в якомусь сенсі є оптимальним [17]. Його результат не може бути поліпшений, тому що у всіх випадках, коли можлива однозначна відповідь, він його дасть - а в тих випадках, коли відповідь неоднозначна, результат кількісно характеризує міру цієї неоднозначності.

Разом з тим, в оптимальності криється і основний недолік ідеального байєсового класифікатора: для його побудови потрібна вибірка, що містить всі можливі комбінації змінних – а розмір такої вибірки експоненційно зростає із

зростанням кількості змінних [18]. Для подолання описаної вище проблеми на практиці використовують наївний байєсів класифікатор – класифікатор, побудований на основі припущення про незалежність змінних, тобто припущення про те, що використання цього припущення дозволяє не вивчати взаємодію всіх можливих поєднань змінних, обмежившись лише впливом кожної змінної окремо на приналежність образу до одного з класів.

Перевагою цього підходу є те, що вимоги до розміру вибірки скорочуються від експоненційних до лінійних. Недолік – те, що модель є точною лише у випадку, коли виконується припущення про незалежність. В іншому випадку, строго кажучи, обчислені ймовірності вже не є точними (і навіть більше того, їх сума може не дорівнювати одиниці, через що потрібно нормувати результат). Однак на практиці незначні відхилення від незалежності призводять лише до незначного зниження точності, і навіть у разі істотної залежності між змінними результат роботи класифікатора продовжує корелювати з істинною приналежністю образу до класів. При цьому переваги класифікатора (висока швидкість роботи, простота і масштабованість, помірні вимоги до пам'яті) часто переважують недоліки.

У теорії ймовірностей та статистиці Теоре́ма Ба́єса (або ж Зако́н Ба́єса, чи Правило Баєса) описує ймовірність події, спираючись на обставини, що могли би бути пов'язані з цією подією [10]. Наприклад, припустімо, що хтось цікавиться, чи має рак певна особа, і знає вік цієї особи. Якщо рак пов'язаний з віком, то, застосовуючи теорему Баєса, інформацію про вік осіб можливо використати для точнішої оцінки ймовірності того, що вони мають рак.

При застосуванні, задіяні у теоремі Баєса ймовірності можуть мати різні інтерпретації. В одній із цих інтерпретацій теорема Баєса використовується безпосередньо у певному підході до статистичного висновування. При баєсовій інтерпретації ймовірності ця теорема виражає, як повинна раціонально змінюватися суб'єктивна міра впевненості при врахуванні свідчення: це є баєсовим

висновуванням, що є фундаментальним для баєсової статистики. Тим не менш, теорема Баєса має численні застосування у широкому спектрі обчислень із залученням ймовірностей, а не лише у баєсовому висновуванні.

Теорема Баєса задається математично таким рівнянням [14]:

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)}$$

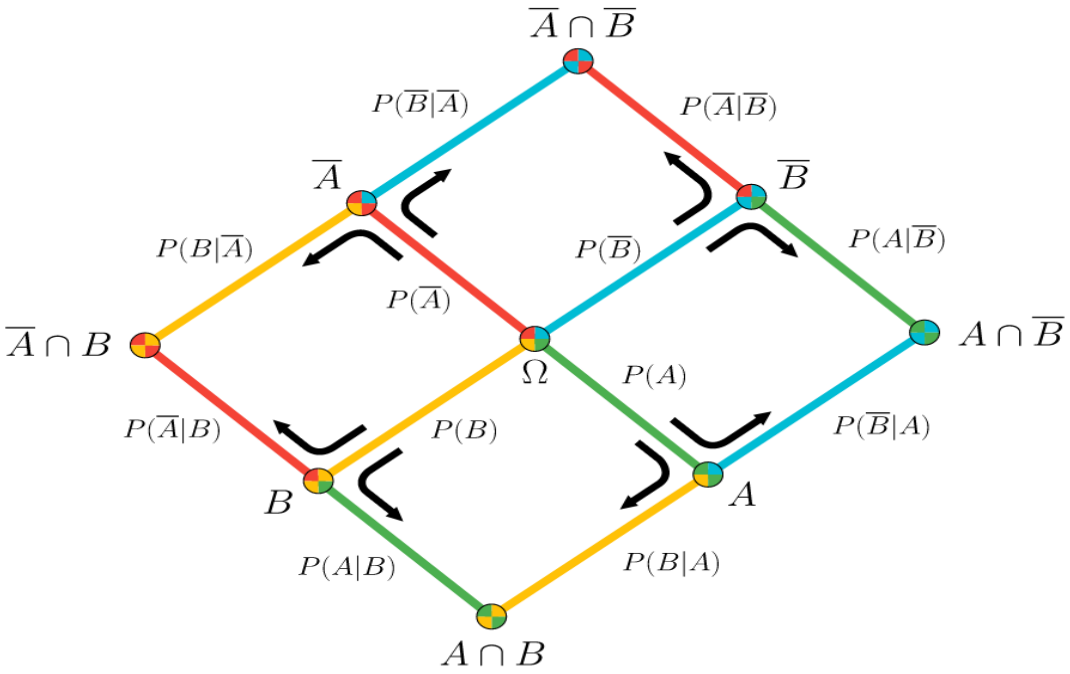
де A та B є подіями:

– P(A) та P(B) є ймовірностями A та B безвідносно одна до одної;

– P(A | B) умовна ймовірність, є ймовірність A за умови істинності B;

– P(B | A) є ймовірністю спостереження події B за умови істинності A.

На рисунку 3.1 відображено візуальна інтерпретація теореми Баєса.



$$P(A|B)\cdot P(B) = P(A\cap B) = P(B|A)\cdot P(A)$$

Рисунок 3.1 – Візуалізація теореми Баєса суперпозицією двох дерев ухвалення рішень

3.2 Метод опорних векторів.

В машинному навчанні метод опорних векторів (ОВМ) – це метод аналізу даних для класифікації та регресійного аналізу за допомогою моделей з керованим навчанням з пов'язаними алгоритмами навчання, які називаються опорно-векторними машинами [12]. Для заданого набору тренувальних зразків, кожен із яких відмічено як належний до однієї чи іншої з двох категорій, алгоритм тренування ОВМ будує модель, яка відносить нові зразки до однієї чи іншої категорії, роблячи це неймовірнісним бінарним лінійним класифікатором. Модель ОВМ є представленням зразків як точок у просторі, відображених таким чином, що зразки з окремих категорій розділено чистою прогалиною, яка є щонайширшою. Нові зразки тоді відображуються до цього ж простору, й робиться передбачення про їхню належність до категорії на основі того, на який бік прогалини вони потрапляють.

Візуальне зображення роботи метода з лінійно роздільними даними [12] показано на рис. 3.3

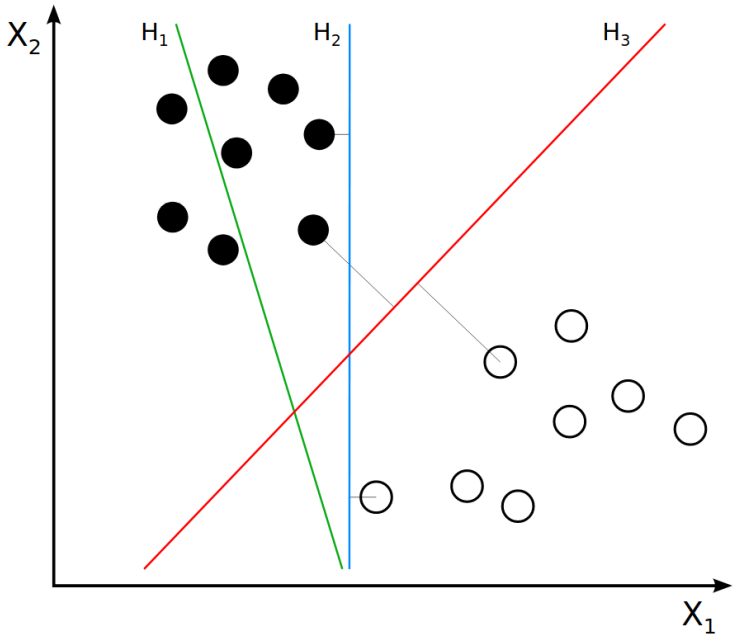


Рисунок 3.3 – Робота метода з лінійно роздільними даними

Як видно з рисунку H_1 не розділяє ці класи. H_2 розділяє, але лише з невеликим розділенням. H_3 розділяє їх із максимальним розділенням.

Але бувають ситуації коли дані неможно лінійно розділити [13]. З цієї причини було запропоновано відображувати первинний скінченновимірний простір до простору набагато вищої вимірності, здогадно роблячи розділення простішим у тому просторі.

Візуальне зображення представлено на рис. 3.4.

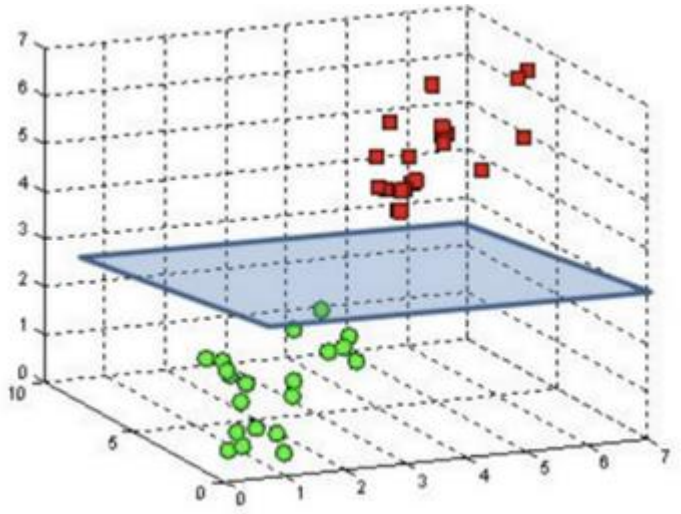


Рисунок 3.4 – Лінійно нероздільні дані

На рис 3.5. можливо побачити візуальне відображення роботи ОВМ.

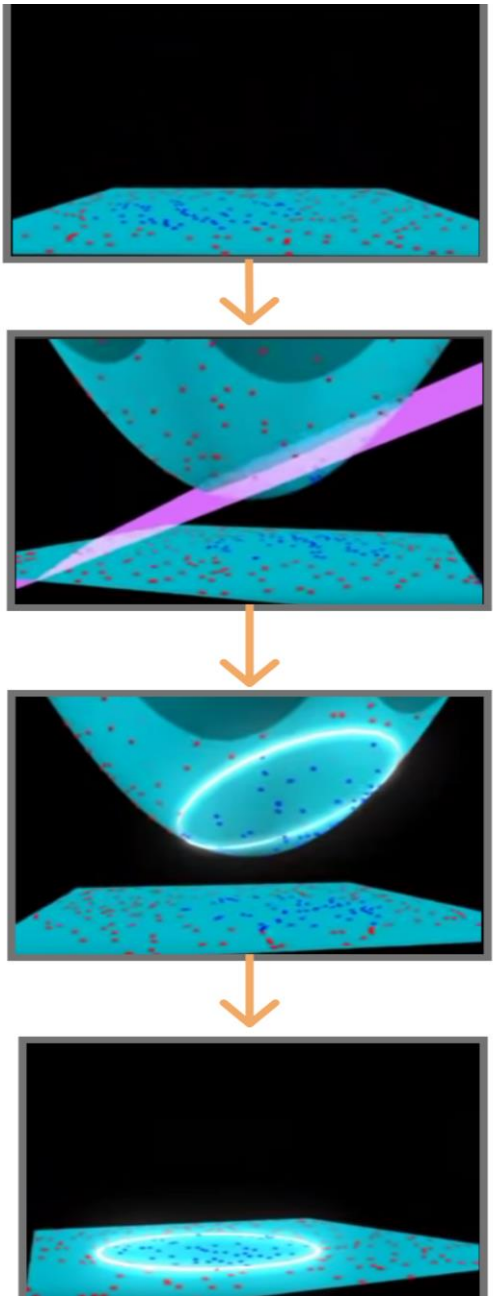


Рисунок 3.5 – Робота метода ОВМ

3.3 Перцептрон.

Перцептро́н – математична або комп'ютерна модель сприйняття інформації мозком (кібернетична модель мозку), запропонована Френком Розенблатом в 1957 році й реалізована у вигляді електронної машини «Марк-1» Перцептрон став однією з перших моделей нейромереж, а «Марк-1» – першим у світі нейрокомп'ютером. Незважаючи на свою простоту, перцептрон здатен навчатися і

розв'язувати досить складні завдання [9]. Основна математична задача, з якою він здатний впоратися – це лінійне розділення довільних нелінійних множин.

Перцептрон складається з трьох типів елементів, а саме: сигнали, що надходять від давачів, передаються до асоціативних елементів, а відтак до реагуючих. Таким чином, перцептрони дозволяють створити набір «асоціацій» між вхідними стимулами та необхідною реакцією на виході. В біологічному плані це відповідає перетворенню, наприклад, зорової інформації у фізіологічну відповідь рухових нейронів

Принцип роботи елементарного перцептрона.

Елементарний перцептрон складається з елементів трьох типів: S-елементів, A-елементів та одного R-елементу. S-елементи – це шар сенсорів, або рецепторів [11]. У фізичному втіленні вони відповідають, наприклад, світлочутливим клітинам сітківки ока або фоторезисторам матриці камери. Кожен рецептор може перебувати в одному з двох станів – спокою або збудження, і лише в останньому випадку він передає одиничний сигнал до наступний шару, асоціативним елементам.

A-елементи називаються асоціативними, тому що кожному такому елементові, відповідає цілий набір (асоціація) S-елементів. A-елемент активізується, щойно кількість сигналів від S-елементів на його вході перевищує певну величину θ.

Сигнали від збуджених A-елементів, своєю чергою, передаються до суматора R, причому сигнал від i-го асоціативного елемента передається з коефіцієнтом wi. Цей коефіцієнт називається вагою A-R зв'язку.

Так само як і A-елементи, R-елемент підраховує суму значень вхідних сигналів, помножених на ваги (лінійну форму). R-елемент, а разом з ним і елементарний перцептрон, видає «1», якщо лінійна форма перевищує поріг θ, інакше на виході буде «-1». Математично, функцію, що реалізує R-елемент, можна записати так:

$$f(x) = sign(\sum_{i=1}^{n} w_i x_i - \theta)$$

Для прикладу було взято формулу з функцією активації сигмоїда, також можливі й інші варіанти.

Навчання елементарного перцептрона полягає у зміні вагових коефіцієнтів wi зв'язків A-R. Ваги зв'язків S-A (які можуть приймати значення (-1; 0; 1)) і значення порогів A-елементів вибираються випадковим чином на самому початку і потім не змінюються.

Після навчання перцептрон готовий працювати в режимі розпізнавання або узагальнення. У цьому режимі перцептрону пред'являються раніше невідомі йому об'єкти, й він повинен встановити, до якого класу вони належать. Робота перцептрона полягає в наступному: при пред'явленні об'єкта, збуджені A-елементи передають сигнал R-елементу, що дорівнює сумі відповідних коефіцієнтів w_i. Якщо ця сума позитивна, то ухвалюється рішення, що даний об'єкт належить до першого класу, а якщо вона негативна – то до другого.

На рис. 3.6 зображено модель елементарного перцептрону.

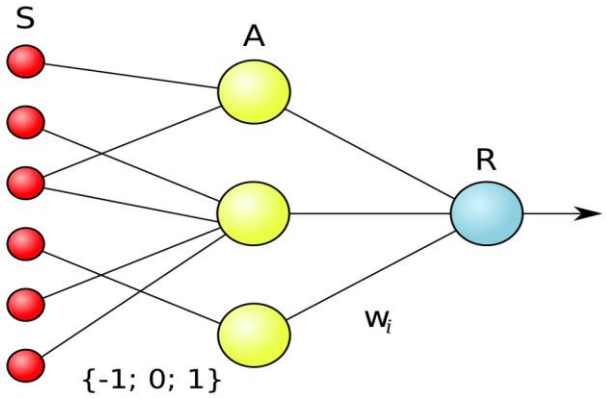


Рисунок 3.6 – Логічна схема елементарного перцептрону

Ваги зв'язків S-A можуть мати значення −1, 1 або 0 (тобто відсутність зв'язку). Ваги зв'язків A-R W можуть мати будь-яке значення. Зазвичай ваги зв’язків генеруються випадково.

РОЗДІЛ 4
РОЗРОБКА ПРОГРАМНОГО ЗАСТОСУВАННЯ ВИЯВЛЕННЯ СПАМ ПОВІДОМЛЕНЬ

Як було сказано раніше, для розробки програмного застосування було використано середовище програмування PyCharm, що надає користувачу можливість легко підключати сторонні бібліотеки [7].

Характеристика обчислювальної машини, що було використано для дослідження наведено на рис. 4.1.

```
system          LIFEBOOK E744
bus             FJNB26F
memory          128KiB BIOS
processor       Intel(R) Core(TM) i7-4702MQ CPU @ 2.
memory          32KiB L1 cache
memory          256KiB L2 cache
memory          6MiB L3 cache
memory          32KiB L1 cache
memory          8GiB System Memory
memory          4GiB SODIMM DDR3 Synchronous 1600 MH
memory          4GiB SODIMM DDR3 Synchronous 1600 MH
bridge          Xeon E3-1200 v3/4th Gen Core Process
```

Рисунок 4.1 – Характеристики обчислювальної машини

Перш ніж подавати на вхід класифікаторам текст, данні повинні бути підготовлені.

4.1 Опис використаного датасету.

В якості навчального датасета був обраний датасет спам повідомлень з сайту kaggle SMS Spam Collection Dataset [17].

На рис. 4.2 датасет містить дві колонки, у першій клас повідомлення (ham – звичайне повідомлення, spam – спам повідомлення). Спочатку перетворимо першу колонку класів на масив цілих чисел. 1 – буде означати що дане повідомлення є спамом, 0 – звичайне повідомлення.

ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat...
ham	Ok lar... Joking wif u oni...
spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std txt rate)T&C's apply 08452810075over18's
ham	U dun say so early hor... U c already then say...
ham	Nah I don't think he goes to usf, he lives around here though
spam	FreeMsg Hey there darling it's been 3 week's now and no word back! I'd like some fun you up for it still? Tb ok! XxX std chgs to send, ◆1.50 to rcv

Рисунок 4.2 – Датасет спам повідомлень

4.2 Перетворення датасету

Тепер для того щоб перетворити увесь текст повідомлень у числа (векторізувати), потрібно скласти словник слів, масив слів [8]. Кожне слово з датасету приводиться до нижнього регістру. Так як у наш набір могли попасти знаки пунктуації, на наступному етапі потрібно очистити словник від непотрібних слів.

Після цього кожне слово приводиться до інфінітиву (риба, рибалка -> риб).

Проаналізувавши отриманий набір даних можна побачити що деякі слова повторюються дуже часто, а інші навпаки, дуже рідко. Видалимо з отриманого набору слова які повторюються більше ніж 1000 та менше ніж 5 разів.

```
mistakeu 1
bornpleas 1
terminatedw 1
inconveni 1
henri 2
yard 1
bergkamp 1
margin 1
itsnot 1
unintent 1
nonetheless 1
hooch 1
toaday 1
splat 1
graze 1
confirmdeni 1
hearin 1
```

Рисунок 4.3 — Слова які рідко зустрічаються

Далі на основі отриманого набору слів створимо словник, видаливши всі повтори (тобто у словнику кожне слово унікальне). Для цього був використаний векторізатор з бібліотеки sklearn.

У створений словник увійшло 1191 слово. Векторізуємо речення датасету за допомогою словника.

Розглянемо даний процес більш детально. Наприклад в нас є словник зі слів:

я, купляти, кіт, собака, вчора, собі.

Тоді речення “Вчора я купив собі собаку” буде векторізовано як:

[1, 1, 0, 1, 1, 1],

а речення “Я купив кота”:

[1, 1, 1, 0, 0, 0].

Після перетворення, дані можна подавати на вхід класифікаторам.

Також деяка частина датасету (приблизно 600 повідомлень), яка не входить до навчальних даних, була відкинута для подальшого тестування алгоритмів.

Тестуючи кожен з алгоритмів, будемо рахувати кількість помилок та помилку розпізнавання.

4.3 Тестування використаних алгоритмів.

Першим протестуємо нашу перцептронну багатошарову нейромережу, яка складається з 4 шарів, а саме: 1 вхідний, 2 схованих та 1 вихідний (рис. 4.4).

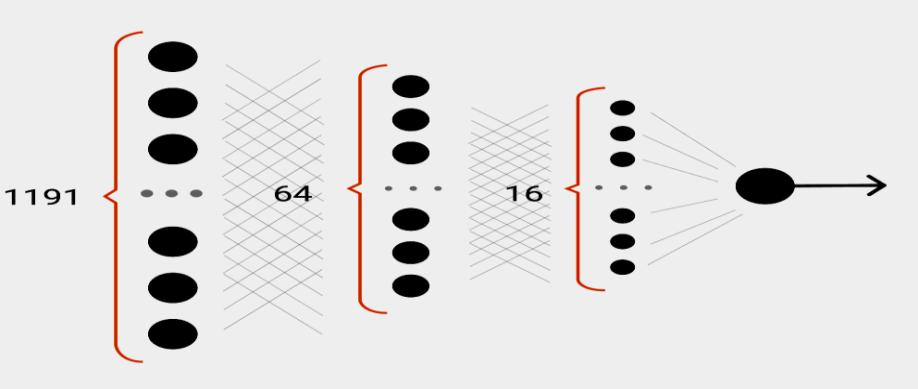


Рисунок 4.4 – Схематична модель перцептронної нейромережі

Протестувавши створену модель отримуємо:

```
Count mistake: 12
Mistake: 2.086957 %
```

Рисунок 4. 5 – Результати роботи перцептронної нейромережі

Як можна побачити на рис. 4.5 середня кількість помилок перцептрона становить 12 з приблизно 600 повідомлень, а середня ймовірність помилки – 2%.

На тих же вхідних даних протестуємо алгоритм на основі наївного байесового класифікатора (рис. 4.6) та методу опорних векторів (SVM) (рис. 4.7).

```
Count mistakes: 10
Mistake: 1.739130
```

Рисунок 4.6 — Результати роботи наївного байєсівського класифікатора

```
Count mistakes: 6
Mistake: 1.043478
```

Рисунок 4.7 – Результати роботи метода опорних векторів

З отриманих даних побудуємо графік наведений на рис. 4.8.

Проведений аналіз показує, що найбільш точним для наведеного датасету спам повідомлень є застосування методу опорних векторів.

Затрати часу на реалізацію досліджуваних методів:

– на навчання перцептрону: 10 епох = 10.11 сек;

5 епох = 6.03 сек;

3 епохи = 4.01 сек;

– наївний байєсівський класифікатор: 0.21 сек;

– метод опорних векторів: 0.31 сек.

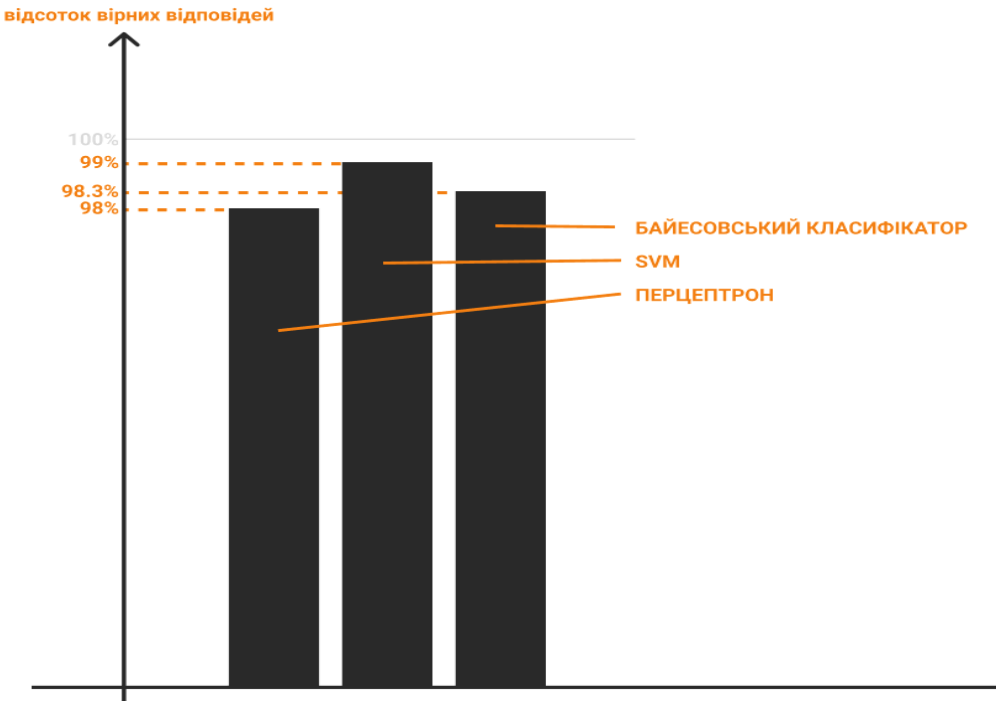


Рисунок 4.8 — Порівняльний графік методів

ВИСНОВКИ

В рамках виконання даної дослідницької роботи було виконано наступні поставлені завдання:

1) був наведений аналіз та дослідження предметної області у сфері розпізнавання спам повідомлень;

2) був виконаний аналіз літературних джерел з питання роботи існуючих методів боротьби зі спамом;

3) був здійснений аналіз порівняння трьох методів розпізнавання спам повідомлень;

4) було виконано обґрунтування використаних програмних засобів розробки;

5) виконана постановка мети та завдання дослідницької роботи;

6) було обґрунтовано актуальність боротьби зі спам повідомленнями;

7) розроблений проект програмного забезпечення;

8) розроблені методи програмної реалізації основних функціональних можливостей.

ПЕРЕЛІК ПОСИЛАНЬ

1. Спам. Види спаму. Боротьба зі спамом. [Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/Спам– Дата доступу: 26.12.2019.

2. Способи поширення спаму. [Електронний ресурс]. – Режим доступу: http://www.refine.org.ua/pageid-5411-1.html – Дата доступу: 26.12.2019.

3. Способи боротьби зі спамом. [Електронний ресурс]. – Режим доступу: http://korysne.ostriv.in.ua/publication/code-24F002FC35B8C/list-1420E79CF27 – Дата доступу: 26.12.2019.

4. Чорні списки. [Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/Чорний_список_(інформатика) – Дата доступу: 26.12.2019.

5. Applications for Python [Електронний ресурс]. – Режим доступу: https://www.python.org/about/apps/ – Дата доступу: 26.12.2019.

6. PyCharm [Електронний ресурс]. – https://www.jetbrains.com/ru-ru/pycharm/ – Дата доступу: 26.12.2019.

7. Коэльо Л.П. Построение систем машинного обучения на языке Python / Л.П. Коэльо, В. Ричарт // – М.: ДМК Пресс, 2016. – 302 с.

8. Brownlee J. Deep learning with python / J. Brownlee // – Jason Brownlee, 2016. – 266 p.

9. Николенко С. Глубокое обучение / С. Николенко, А. Кадурин, Е. Архангельская // – СПб.: Питер, 2018. – 480 с.

10. Метод опорних векторів [Електронний ресурс]. – Режим доступу: http://om.univ.kiev.ua/users_upload/15/upload/file/pr_lecture_07.pdf – Дата доступу: 26.12.2019

11. Метод опорних векторів – введення в алгоритми машинного навчання. [Електронний ресурс]. – Режим доступу: https://trainmydata.com/article/mietod-

opornykh-viektorov-vviedieniie-v-alghoritmy-mashinnogho-obuchieniia – Дата доступу: 26.12.2019.

12. Формула повної імовірності. Формула Байєса. [Електронний ресурс]. – Режим доступу: https://web.posibnyky.vntu.edu.ua/fitki/4tichinska_teoriya_jmovirnostej /17.htm – Дата доступу: 26.12.2019.

13. W. Zhang and F. Gao, "Performance analysis and improvement of naïve Bayes in text classification application", IEEE Conference Anthology, China, 2013, pp. 1-4. doi: 10.1109/ANTHOLOGY.2013.6784818.

14. B. Liu, E. Blasch, Y. Chen, D. Shen and G. Chen, "Scalable sentiment classification for Big Data analysis using Naïve Bayes Classifier", 2013 IEEE International Conference on Big Data, Silicon Valley, CA, 2013, pp. 99-104. doi: 10.1109/BigData.2013.6691740.

15. A. McCallum and K. Nigam, "A Comparison of Event Models for Naive Bayes Text Classification", Learning for Text Categorization: Papers from the 1998 AAAI Workshop, pp. 41-48.

16. S.D. Sarkar, S. Goswami, A. Agarwal and J. Aktar "A Novel Feature Selection Technique for Text Classification Using Naive Bayes", Hindawi Publishing Corporation International Scholarly Research Notices Vol. 2014, Article ID 717092, 10 p. doi: 10.1155/2014/717092.

17. Датасет спам повідомлень. [Електронний ресурс]. – Режим доступу: https://www.kaggle.com/uciml/sms-spam-collection-dataset – Дата доступу: 26.12.2019.

18. Шифр «АнтіСпам». Аналіз емоційного забарвлення текстів та мовлення / Шифр «АнтіСпам» // Проблеми інформатизації: Тези доп. VI міжнародної науково-практичної конференції (13-15 листопада 2019 р.). – Черкаський державний технічний університет, 2019 – С.52.

19. Шифр «АнтіСпам». Распознавание концептов эмоций в лингвистическом процессоре экспертной системы / Шифр «АнтіСпам», Н.Ю. Любченко,

Ю.Ю. Шамаєва // Системи обробки інформації. – Харків: ХУ ПС. – вип. 1(82). – 2010. – С. 8 – 12.

20. Шифр «Пожежна безпека». Research on spam detection using different methods of spam detection / Шифр «Пожежна безпека» // Системи управління навігації та зв’язку. – Полтавський національний технічний університет імені Юрія Кондратюка. - №. 1 (59). – 2020. – С.102-105.