УДК 004.912 DOi 10.30837/ bi.2021.1(96).04

[1]В.О.Нечіпор, [2]А.Л.Єрохін
[1]ХНУРЕ, Україна, volodymyr.nechipor.cpe@nure.ua
[2]ХНУРЕ, Україна, andriy.yerokhin@nure.ua, ORCID iD: 0000-0002-8867-889X

МОДИФІКАЦІЯ МЕТОДУ КЛАСИФІКАЦІЇ БАЙЄСА В ЗАДАЧАХ ВИЯВЛЕННЯ СПАМУ УКРАЇНСЬКОЮ МОВОЮ

Стаття присвячена аналізу існуючих технологій для виконання задачі класифікації української мови з метою фільтрації спаму. В рамках дослідження було проаналізовано недоліки методу класифікації Байєса в рамках сучасної реалізації цього методу на мові програмування Python для роботи з українською мовою. Основним недоліком програмної реалізації методу Байєса було виявлено некоректний для української мови поділ на слова за умови, що слова містять апостроф. Для виправлення цієї проблеми було розроблемо модифікований метод класифікації за Байєсом, який коректно працює зі словами української мови, що містять апостроф. В результаті вдалось підняти ефективність спрогнозованого класифікування спаму з 86% до 91%.

МАШИННЕ НАВЧАННЯ, МЕТОД КЛАСИФІКАЦІЇ БАЙЄСА, УКРАЇНСЬКА МОВА, АПОСТРОФ, СПАМ, ПРОГНОЗУВАННЯ

Нечипор В.А., Ерохин А.Л. Модификация метода классификации Байеса в задачах обнаружения спама на украинском языке. Статья посвящена анализу технологий решения задачи классификации украинского языка с целью фильтрации спама. В рамках исследования были проанализированы недостатки метода классификации Байеса в рамках современной реализации этого метода на языке программирования Python для работы с украинским языком. Главным недостатком программной реализации метода Байеса был выбран некорректный для украинского языка разбор на предложения на слова при условии, что слова содержат апостроф. Для исправления этой проблемы был разработан модифицированный метод классификации по Байесу, который корректно разбивает предложения на слова, даже при наличии апострофов. В результате удалось поднять эффективность прогнозированной классификации спама с 86% до 91%.

МАШИННОЕ ОБУЧЕНИЕ, МЕТОД КЛАССИФИКАЦИИ БАЙЕСА, УКРАИНСКИЙ ЯЗЫК, АПОСТРОФ, СПАМ, ПРОГНОЗИРОВАНИЕ

Nechipor V.O., Yerokhin A.L. Modification of the Bayes classification method in the tasks of detecting spam in the Ukrainian language. Existing technologies for the task of Ukrainian language classification with in order to filter spam are analyzed in this article. As part of the study, the shortcomings of the Bayesian classification method towards Ukrainian language were analyzed in the context of the modern implementation of this method in Python programming language. The main disadvantage of the software implementation of the Bayesian method was incorrect for the Ukrainian language parsing of sentences into words, provided that the words contain apostrophes. To correct this problem, a modified Bayesian classification method was developed, which correctly splits sentences into words, even if they contain apostrophes. As a result, the efficiency of the predicted spam classif ication was raised from 86% to 91%.

MACHINE LEARNING, BAYES CLASSIFICATION METHOD, UKRAINIAN LANGUAGE, APOSTROPHE, SPAM, FORECASTING

Вступ

Життя в сучасному світі тісно пов'язане з онлайн присутністю – Інтернет є джерелом новин, соціальною мережею, архівом інформації про будь-що з історії, робочою платформою, інструментом для освіти, тощо. У 2020-му році більш ніж 4.5 мільярди людей постійно користувались Інтернетом, а це більша половина населення планети [1].

Невід'ємною частиною онлайн присутності є електронна пошта, яка використовується для ідентифікації користувача в мережі і листування, отримання звітів і чеків про купівлю в мережі. Згідно останньою статистики щоденно відправляється 319 мільярдів листів [2], з яких, очевидно, далеко не всі є бажаними. Електронна пошта є не тільки зручним інструментом, але й небезпекою – величезна кількість небажаних листів щоденно розсилається багатьом користувачам, чия електронна адреса стала відома тим, хто розсилає їх. Саме феномен відправки небажаних листів називається спамом, а зміст листів може бути як просто набридливою рекламою, так і погрозами, спробами обманути і заманити на не доброчесний сайт, перейти за фішинговим посиланням, розсилкою з інформацією про незаконні ресурси: порно, жорстокі веб-сайти, тощо. Феномен спаму набув нечуваних масштабів у світі, що робить розробку сервісів захисту від спаму надзвичайно актуальним завданням.

В основі будь-якої програми захисту від спаму (спам-фільтрів) лежить мовний аналіз, а саме алгоритм, який зможе класифікувати повідомлення за бінарним критерієм «спам»/«не спам». Необхідність мовного аналізу ускладнює створення спам-фільтрів тим, що різні мови належать до різних груп мов, відрізняються граматичними і семантичним правилами побудови речень, абеткою, містять унікальні допоміжні символи (як апостроф в українській мові).

На сьогодні існують готові рішення для спам-фільтрів найбільш популярних мов, в першу чергу для англійської, як для найпоширенішої мови світу. В той же час український сектор лише починає

розвиватись в напрямку створення сервісів, які вміють досконало аналізувати українську мову.

Спам-фільтр є одним зі складних прикладів програмного забезпечення і вимагає застосування машинного навчання. В рамках дослідження розроблено нейронну мережу з вчителем.

У даній роботі проаналізовано методи класифікації текстів з метою фільтрування спаму українською мовою. Для виконання задачі дослідження обрано метод наївної класифікації Байєса як метод, який, на думку дослідників, найкраще підходить для даної мети.

Для програмної реалізації методу Байєса було обрано Python, як мову з найбільшою історією роботи з машинним навчанням. Саме на Python були розроблені одні з перших нейронних мереж, що слугувало поштовхом подальшого розвитку інфраструктури саме цією мовою.

Оскільки перші кроки для створення програмного забезпечення машинного навчання з аналізу української мови потребують якомога більшої підтримки, було вирішено, що Python і його сучасна база буде кращим рішенням. Під час написання програми було використано найбільш розповсюджену бібліотеку з уже реалізованим методом Байєса — nltk. В процесі реалізації методу наївної класифікації Байєса для класифікації текстів українською мовою на мові Python було знайдено перепону, яка заважала правильно класифікувати деякі повідомлення за ознакою «спам»/«не спам», а саме - український апостроф, який на відміну від англійського апострофа, часто є невід'ємною частиною кореня слова, а не способом скорочення речення шляхом поєднання двох різних слів. Внаслідок цього підготовка тексту до класифікації за Байєсом некоректно розділяла речення на слова і призводила до хибних висновків.

1. Постановка задачі

Об'єктом даного дослідження є мовний аналіз текстів з метою їх подальшої класифікації, а предметом дослідження – методи класифікації текстів українською мовою.

Основним завданням роботи є розробка модифікованого методу наївної класифікації Байєса з метою покращення результатів обробки листів українською мовою. Виділимо наступні завдання, які необхідно виконати в рамках дослідження:

1) проаналізувати аналіз методів класифікації текстів;

2) розробити програмне забезпечення спам-фільтру українською мови з використанням методу наївної класифікації Байєса;

3) розробити модифіковане програмне забезпечення спам-фільтру українською мови з використанням методу наївної класифікації Байєса, яке коректно розділятиме українське речення на слова;

4) порівняти дві реалізації.

2. Аналіз задач та методів класифікації тексту

Проаналізуємо основні методи класифікації текстів, розберемо їхні особливості з точки зору придатності для використання в методах фільтрації спаму українською мовою. Методи класифікації текстів лежать на межі двох областей – машинного навчання та інформаційного пошуку [3]. В рамках даного дослідження в першу чергу аналізуватимемо саме автоматичну класифікацію текстів, адже саме вона потрібна для реалізації програмного забезпечення спам-фільтру українською мови. Для успішного функціонування система автоматичної класифікації потребує розробок на основі машинного навчання, а саме – створення нейронної мережі з вчителем. У випадку створення програмного забезпечення спам-фільтру потрібна наявність бази даних листів чи текстів які вже класифіковані за категорією «спам»/«не спам». При цьому враховуємо, що автоматична класифікація складається з двох етапів: навчання моделі і її використання.

На етапі навчання моделі необхідно виконати опис множини заздалегідь визначених категорій і представлення тренувального набору елементів з уже визначеною категорією. Модель алгоритму побудує власні правила класифікації на основі певної тренувальної вибірки даних, які ми їй запропонуємо. Саме крок підбору даних для тренування моделі є найбільш важливим для правильного функціонування моделі, адже неправильно класифіковані дані або їх дуже мала кількість можуть спричинити збої в роботі моделі. Використання моделі полягає у визначенні категорій нових, раніше невідомих даних. Модель вважається успішною, якщо кількість правильно здійснених класифікацій перевищує 50%.

Задачу класифікації можна представити як необхідність визначити множину, до якої належить певний текст українською мовою, або необхідність знайти функцію, яка буде максимально наближена до ідеалу. Кожному елементу t ставиться у відповідність набір ознак $t = \{X_i\}$. Далі застосовується алгоритм класифікації для виділення текстів, найбільш відповідних заданому класу [4].

Для класифікації застосовуються різноманітні методи, кожен з яких має свої переваги і особливості використання. Переважна більшість методів класифікації текстів так чи інакше засновані на припущені, що тексти, які відносяться до однієї категорії, мають однакові ознаки (слова чи словосполучення), і наявність чи відсутність таких ознак в тексті визначає його приналежність чи неприналежність до тієї чи іншої теми. Таким чином, для кожної категорії повинна бути множина ознак, яку називають словником, через те, що вона складається з лексем, які включають слова та/або словосполучення, що характеризують категорію. Подібно категоріям, кожен текст також має ознаки, за якими його можна

віднести з деяким ступенем вірогідності до однієї чи декількох категорій. Множина ознак усіх текстів повинна співпадати з множиною ознак категорій.

Задача методів класифікації текстів полягає в тому, щоб якнайкраще обрати такі ознаки і сформулювати правила, опираючись на які й буде прийматися рішення щодо віднесення документа до рубрики. Найбльш відомими з цих методів є [5]: класифікація через дерево рішень; Байєсівська (наївна) класифікація; класифікація за методом опорних векторів; класифікація за методом найближчого сусіда; класифікація штучними нейронними мережами. Для подальшого дослідження і вдосконалення оберемо метод наївної класифікації Байєса.

Наївний Байєсівський класифікатор традиційно використовується в задачах класифікації текстів, таких як фільтрація спаму, автоматична рубрикація або визначення тональності документа. Набули поширеного розвитку два його різновиди: багатомірний (multivariate) та мультиноміальний (multinomial). У загальному вигляді визначають найбільш вірогідний клас алгоритмом наївної байєсівської класифікації; класифікація зводиться до обчислення максимального значення аргументу, при відомому наборі незалежних ознак $x_1, x_2, \supset x_n$.

Класифікація тексту при цьому виглядає так:

$$C(T) = \max \sum (t_1, t_2, \ldots t_n | C),$$

де T – текст, що класифікується, а $t_1, t_2, \supset t_n$ – набір речень тексту. Так, приналежність тексту до тієї чи іншої категорії зводиться до обчислення максимального значення суми коефіцієнтів приналежності реченням [6].

Зазначений метод використовує ймовірнісну модель, в якій класифікація та включення у відповідну категорію документів проводиться шляхом оцінювання ймовірності появи слів у документі. Ймовірності можуть бути використані для оцінки найбільш близьких категорій тестового документа [7].

Метод класифікації Байєса має декілька різних варіацій, кожна з яких краще підходить для вирішення тієї чи іншої задачі. Наївний класифікатор найкраще себе показує тоді, коли певну множину об'єктів потрібно класифікувати – розбити на класи. Однак наївний класифікатор менш ефективний, коли логіка розподілення має бути трішки складнішою. Мережевий класифікатор Байєса, наприклад, дозволить спочатку згрупувати множину об'єктів за проміжними спільними ознаками і знайти залежність класу від однієї чи групи ознак. Основні переваги наївного байєсівського класифікатора – це простота реалізації і низькі обчислювальні витрати при навчанні та класифікації. У порівнянні, наприклад, з методом k-найближчого сусіда, який пропонує порівнювати аналізований документ з усіма документами з навчальної вибірки і тому вимагає тривалого часу роботи, алгоритм Байєса швидший в десятки разів [8]. Основним недоліком методу є відносно невисока якість класифікації в більшості реальних завдань [8]. Зазначений метод часто використовується як базовий метод при порівнянні різних методів машинного навчання.

3. Датасет спам фільтру

Задача фільтрації спаму не нова, найбільшого розвитку вирішення цієї проблеми досягло саме в рамках англійської мови, як однієї з найбільш популярних мов в Інтернет кореспонденції. На веб-сайті kaggle.com, наприклад, легко знайти чіткі інструкції з налаштування спам-фільтру [9], а також готові дата сети текстів, які вже класифіковані за принципом «спам»/«не спам».

Збір, аналіз і класифікація такого датасету є окремою масштабною задачею, тому його наявність конкретною мовою є важливим елементом для підготовки в роботі зі спам-фільтром. В рамках роботи над поставленою задачею не було виявлено доступних датасетів спам текстів українською мовою. Це підтверджує актуальність даного дослідження і створює певні складнощі для досягнення мети роботи.

Для продовження роботи над аналізом методів класифікації текстів було зібрано власний датасет відносно невеликого обсягу розміром в декілька сотень текстів. Частина текстів була зібрана з поштових ящиків і смс-повідомлень. Оскільки мета спам-повідомлень різною мовою приблизно однакова – повідомити небажані новини або спробувати обманути і виманити дані – знехтуємо тим, що смс-повідомлення і електронні листи часто дещо відрізняються об'ємами тексту, що передається.

Метадані повідомлення в рамках цього дослідження також не бралися до уваги (інформація щодо адреси поштової скриньки відправника, сайту відправника, рейтингу доброчесності відправника, заголовка листа, тощо). Предметом дослідження стало власне тіло листа українською мовою. Зібраний датасет обсягом 356 текстових повідомлень, що включають 30% спам повідомлень і 70% звичайних повідомлень було оформлено в такому вигляді (табл.1):

Таблиця 1

Датасет повідомлень українською

Label	Body
spam	МЕГА РОЗПРОДАЖ! Знижки -70%+ додатково -20%: http://bit.ly/**
ham	Добрий день, гадаю, в такому вигляді можна прийняти звіт

Спам повідомлення отримали маркування «spam», а звичайні повідомлення – «ham». У сфері спам фільтрів досить давно ввели поняття «ham» як коротке і співзвучне зі спамом для маркування повідомлень, що не являються спамом. Адже кожного разу використовувати «не спам» досить не зручно. Дотримуватимемось загальноприйнятої тенденції в цьому дослідженні.

4. Практична реалізація метода Байєса

Для практичної реалізації методу Байєса використаємо сервіс Datalore від компанії Jetbrains, який надає безкоштовний доступ до командої оболонки для інтерактивних розрахунків Jupiter Notebook. Спочатку спробуємо використати уже розроблений мультиноміальний метод Байєса з бібліотеки nltk. На першому кроці випадковим чином розіб'ємо датасет на частину для навчання і частину для перевірки, де 80% усіх повідомлень будуть використані для навчання. Після цього навчаємо модель на основі тренувального датасету і проводимо тестування. Метод включає в себе наступні кроки:

1) прибираємо дублікати повідомлень;

2) прибираємо пунктуацію;

3) трансформуємо всі літери в нижній регістр;

4) розбиваємо речення на слова;

5) прибираємо стоп-слова (слова, які не мають важливого значення, коли відірвані від контексту).

6) вираховуємо вагу кожного слова відносно категорії (залежить від частоти використання в певній категорії в навчальному датасеті);

7) перевіряємо результат на тестовому датасеті.

В даному дослідженні ефективність роботи класифікаторів за Байєсом вимірювалась за шкалою від нуля до одиниці, де 1 це 100% правильно класифікованих повідомлень, а 0 – жодного правильно класифікованого повідомлення. У результаті фільтрування спаму отримуємо результат 0.86 (рис. 1). Це показник ефективності, який частково залежить від невеликого обсягу датасету і від того, що весь датасет отримано з поштової скриньки однієї людини.

```
              precision    recall  f1-score   support

           0       0.76      1.00      0.87        13
           1       1.00      0.75      0.86        16

    accuracy                           0.86        29
   macro avg       0.88      0.88      0.86        29
weighted avg       0.89      0.86      0.86        29

Confusion Matrix:
 [[13  0]
 [ 4 12]]

Accuracy:  0.8620689655172413
```

Рис. 1. Ефективність мультиноміального методу Байєса

Звернімо увагу на те, що всі 100% листів не були класифіковані правильно. Проаналізуймо помилкові приклади і спробуємо знайти закономірність, притаманну українській мові. Серед хибно класифікованих повідомлень знайдемо декілька прикладів з словами, що містять апострофи і висуваємо теорію, що саме це може бути причиною хибного висновку.

Таблиця 2

Хибно класифіковані повідомлення

Label	Body	Prediction
spam	Спекотна п'ятниця! 50% на все. Тільки 3 дні.	ham
spam	М'язиста пропозиція! Дев'ять днів тренування за кошт п'яти. ФК Плутон на Широнінців.	ham
ham	Припиню зв'язок з сім'єю на декілька днів доки не з'ясую свою ситуацію із здоров'ям.»	spam

Пояснимо причину такого висновку на першому прикладі, відслідкувавши трансформацію повідомлення протягом семи описаних кроків трансформації тексту.

Прибираємо пунктуацію: *«Спекотна п ятниця 50 на все Тільки 3 дні»*.

Трансформуємо всі літери в нижній регістр: *«спекотна п ятниця 50 на все тільки 3 дні»*.

Розбиваємо речення на слова: *[‘спекотна’, ‘п’, ‘ятниця’, ‘50’, ‘на’, ‘все’, ‘тільки’, ‘3’, ‘дні’]*.

Прибираємо стоп-слова (слова, які не мають важливого значення, коли відірвані від контексту): *[‘спекотна’, ‘все’, ‘тільки’, ‘дні’]*.

Вираховуємо вагу кожного слова відносно категорії (залежить від частоти використання в певній категорії в навчальному дата сеті). Кількість слів у реченні, яке ми розглядаємо, невелика: після видалення слова «п'ятниця» залишилось лише чотири слова (табл. 3).

Таблиця 3

Розбір прикладу речення на оцінені слова

Слово	Кількість використань в спамі	Кількість використань не в спамі
‘спекотна’	0	0
‘все’	21	23
‘тільки’	6	4
‘дні’	8	10

Наступним кроком за допомогою алгоритма оцінена кількість вживань даних слів у повідомленнях, промаркованих як «спам» і «не спам» у тренувальному датасеті. Кількість вживань у повідомленнях з позитивною конотацією переважає у двох з чотирьох слів, а одне із слів не вживалось в жодному з повідомлень обраного тренувального дата сету. Це лише початкова обробка тексту після якої необхідна обробка даних алгоритмом Байєса, однак цих даних вже достатньо, щоб зрозуміти, що алгоритму надається некоректна інформація, що веде до хибного висновку.

Як уже було зазначено, слово «п'ятниця» було поділене на «п» і «ятниця», тому що стандартний механізм поділу, розроблений з огляду на англійську мову, замінює апостроф на пробіл, і далі ділить по ньому.

При фінальній класифікації текстів вага кожного слова в категоріях «spam» і «ham» дуже важлива і може схилити класифікатор на одну чи іншу сторону. Якби слово «п'ятниця» було розпізнане вірно, воно мало б високий негативний коефіцієнт ймовірності бути в спамі, тому що лише серед підібраного невеликого дата сету було декілька повідомлень про «Чорну п'ятницю» і супутні знижки. Оскільки слово розпізнане не було, то його коефіцієнтом знехтували, а слова, що залишились, були недостатньо переконливими для вірної класифікації.

Схожа ситуація і з іншими прикладами – багато з них були класифіковані некоректно через те, що аналізатор зміг обробити лише частину слів через неправильний поділ речення на слова.

Отже, після аналізу речення *«Спекотна п'ятниця! 50% на все. Тільки 3 дні.»* зробимо висновок, що наявність апострофів в словах української мови є перепоною для роботи алгоритму класифікації Байєса. Серед хибно класифікованих речень більшість містять апострофи, які, якщо і не є головною причиною хибного висновку, є причиною неможливості коректно класифікувати конкретні слова. Це, в свою чергу, призводить до зменшення кількості слів, які могли б схилити алгоритм на одну із сторін класифікації – «спам»/«не спам».

5. Реалізація методу наївної класифікації Байєса

Метод мультиноміальної класифікації Байєса показав результат в 86% ефективності класифікації повідомлень на «spam» і «ham». Оскільки саме варіації методу Байєса в вирішенні задачі класифікації повідомлень на дві основні категорії вважаються найбільш ефективними, порівняємо метод мультиноміальної класифікації з методом наївної класифікації Байєса. Для подальшого аналізу було реалізовано метод наївної класифікації Байєса для того щоб була можливість модифікувати його частини з метою усунення проблеми виявлення слів з апострофами.

Спочатку завантажуємо тренувальну базу до алгоритму і виконуємо приведення до нижнього регістру і прибирання пунктуації. В першій реалізації методу наївної класифікації Байєса жодної оптимізації по роботі з апострофами не очікується. Спочатку порівняймо який із варіантів методу краще впорається з задачею класифікації тексту на «spam» і «ham».

Одразу після виконання частини програмного коду сервіс Datalore дозволяє вивести частину результату для візуалізації. Було виведено три останні екземпляри листів з тренувального датасету. Лише серед початків речень трьох екземплярів видно проблемне місце зі словом «кур'єр», яке було трансформовано в «кур» і «єр» і, відповідно, не буде вірно проаналізоване з точки зору використання в листах спаму і бажаних листах.

Наступним кроком реалізації алгоритму є розбиття речення на слова і підрахунок кількості використання слів в конкретних реченнях включаючи асоціацію з відповідним типом повідомлення: «spam» і «ham». Результат виконання цієї частини наведено у табл. 4, яка представлена у вигляді матриці; в ній наведено приклад на основі трьох колонок, а отже трьох унікальних слів.

Таблиця 4

Розбір прикладу речення на оцінені слова

label	body	підозра	єр	витрачайте
ham	[«є», «підозра», «що», «ловити» ...]	1	0	0
spam	[«володимире», «не», «витрачайте», «кошти» ...]	0	0	1
ham	[«вітаємо», «кур», «єр», «в», «дорозі» ...]	0	1	0

Повна кількість колонок дорівнює кількості унікальних слів в повідомленнях. Значення нуля чи одиниці означає відповідно відсутність і використання слова в повідомлені, що вказане в даному рядку.

На даному етапі можна вважати, що навчання моделі виконано, за умови що вищевказану класифікацію окремих слів було пройдено на датасеті великого розміру.

Оскільки готового класифікованого датасету спам повідомлень українською мовою о не було, обмежимось кількістю повідомлень в 364 одиниці, які були зібрані власноруч.

Для навчання моделі використаємо приблизно 80% датасету, що складає 293 повідомлення. Повідомлення, що залишились, розміром в 71 одиницю, будуть використані для перевірки тренованої моделі. Результат роботи наївного класифікатора Байєса на тому ж самому датасеті, який було використано в реалізації мультиноміального методу, наведено на рис. 2.

```
Correct: 60
Incorrect: 11
Accuracy: 0.8450704225352113
```

Рис. 2. Ефективність наївного класифікатору за Байєсом

Отже ефективність наївного класифікатору за Байєсом складає 84%, тобто 84% повідомлень були класифіковані вірно. Це на 2% менше ніж за мультиноміальним методом класифікації Байєса. Невеликий відсоток різниці обумовлений, перш за все, відносно невеликим розміром тестового дата сету. Наявність різниці дозволяє нам припустити, що мультиноміальний метод класифікації Байєса є більш ефективним для виконання задачі класифікації листів на «spam» і «ham» за умови наявності більшого тренувального дата сету. Проте і мультиноміальний і наївний методи підходять для успішного виконання задачі класифікації спаму.

Проведемо статистичне порівняння ефективності методів класифікації. Оскільки кожен з методів класифікації було використано лише один раз, ми не можемо бути достеменно впевнені в коректності наведених результатів. Існує декілька основних підходів до поділу датасету на тестовий і тренувальний під час навчання моделі:

Свідомо використовуємо один і той самий набір тренувальних текстів та не міняємо їх між використаннями алгоритму. Перевага: дозволяє уникнути непередбачуваності тренувальних даних після першої обробки. Недолік: за умов неправильно підібраного тренувального датасету, модель приречена на провал, адже у неї немає шансу перевчитись.

Випадково розбиваємо набір даних на тренувальні і тестові під час кожного навчання моделі у розробці. Перевага: усуває проблему першого підходу – модель може перенавчитись, якщо в одному з попередніх використань було підібрано невірний датасет. Недолік: випадковість результату ефективності класифікації викликає недовіру до одного отриманого результату, який може бути «випадково» успішним.

Під час попередніх вимірів ефективності мультиноміального і наївного класифікаторів за Байєсом був використаний другий підхід з випадковістю розподілення даних на тестові і тренувальні. Для того, щоб впевнитись в отриманих результатах і усунути або хоча б мінімізувати вплив випадковості результатів, збережемо результат виконання кожного з алгоритмів щонайменше десять разів і знайдемо середню ефективність. Результат навчання моделей наведено у табл. 5.

Таблиця 5

Середнє арифметичне ефективності мультиноміального і наївного методів класифікації

Спроба	Мультиноміальний	Наївний
1	0,862	0,845
2	0,794	0,816
3	0,884	0,810
4	0,826	0,873
5	0,869	0,788
6	0,783	0,845
7	0,873	0,802
8	0,836	0,760
Спроба	**Мультиноміальний**	**Наївний**
9	0,921	0,901
10	0,894	0,830
Середнє	**0,854**	**0,827**

Середнє арифметичне результатів ефективності використання мультиноміального методу класифікації Байєса складає 0,854, а середнє арифметичне результатів методу наївної класифікації Байєса 0,827. Різниця ефективності збереглась, і навіть збільшилась на користь мультиноміального методу Байєса з 0,017 до 0,027.

На основі проведеного дослідження можемо зробити остаточний висновок, що мультиноміальний метод класифікації Байєса є більш ефективним, ніж метод наївної класифікації для задачі класифікації спаму листів українською мовою.

Під час дослідження обох методів класифікації було помічено проблему розподілення слів речення на слова у зв'язку з українськими апострофами, які, на відміну від англійських апострофів, є невід'ємною частиною слова.

З метою подальшого вдосконалення способів вирішення задачі класифікації текстів з метою фільтрування спаму українською мовою внесемо модифікацію, пов'язану з вирішенням проблеми апострофів, в метод наївної класифікації, який вже було реалізовано.

6. Реалізація модифікованого наївного методу класифікації Байєса

Модифікуємо частину коду, яка відповідає за розбиття речення на слова. Для цього потрібно змінити налаштування пунктуаційних символів, щоб апостроф перестав вважатись символом пунктуації і став вважатись частиною слова. Також необхідно впевнитись, що внесені зміни не зменшують ефективність інших частин алгоритму. Переглянемо символьну таблицю і визначимо символи пунктуації, які необхідно прибрати з речення українською мовою. Далі використаємо модифіковану частину програмного коду для роботи з пунктуаційними символами в методі наївної класифікації Байєса. При розбитті речень на слова після модифікації отримуємо:

1. [‘на’, ‘ваше’, ‘ім’я’, ‘надійшли’, ‘кошти’, ‘для’, ‘отримання’, ‘дзвоніть’, ‘080****305’, ‘код’, ‘22’];
2. [‘спекотна’, ‘п’ятниця’, ‘50’, ‘на’, ‘все’, ‘тільки’, ‘3’, ‘дні’].

Слова з апострофами тепер сприймаються правильно і будуть мати коректний коефіцієнт ймовірності спаму. Це особливо суттєво в невеликих повідомленнях, де може бути лише декілька навантажених важливим значенням слів. Якщо якусь частину з них просто прибрати так само як «стоп-слова» то класифікація найбільш ймовірно буде проведена хибно. Виміряємо ефективність модифікованого методу на тому ж дата сеті, що й класичні методи мультиноміальної та наївної класифікації. Результат наведено на рис. 3.

```
Correct: 65
Incorrect: 6
Accuracy: 0.9154929577464789
```

Рис. 3. Ефективність Модифікованого Методу Байєса

Таким чином, модифікація однієї частини методу класифікації Байєса внесла зміни в фінальний прогноз і дозволила вірно класифікувати слова, специфічні для української мови. Відзначимо, що результат модифікованого методу наївної класифікації вищий,

ніж результат мультиноміального класифікатору. Для повноти аналізу проведемо статистичне дослідження ефективності модифікованого методу і визначимо середнє арифметичне ефективності за 10 тренувань алгоритму. Результат наведено у табл. 6.

Таблиця 6

Середнє арифметичне ефективності мультиноміального, наївного і модифікованого наївного методів класифікації

Спроба	Мультиноміальний	Наївний	Модифікований Наївний
1	0,862	0,845	0,915
2	0,794	0,816	0,901
3	0,884	0,810	0,830
4	0,826	0,873	0,915
5	0,869	0,788	0,845
6	0,783	0,845	0,887
7	0,873	0,802	0,920
8	0,836	0,760	0,887
9	0,921	0,901	0,901
10	0,894	0,830	0,830
Середнє	**0,854**	**0,827**	**0,883**

Середнє арифметичне ефективності класифікованого спаму листів українською мови модифікованого методу наївної класифікації Байєса складає 0,883. Це на 0,561 або 5,6% більше, ніж метод наївної класифікації до внесення змін. Отже, вдалось підвищити ефективність наївного класифікатору Байєса для задачі класифікації листів українською мовою на 5,6%, окрім цього, різниця з середнім арифметичним ефективності мультиноміального методу класифікації складає 0,289 або 2.9%.

Для того, щоб остаточно впевнитись в ефективності внесених змін, протестуємо оригінальний і модифікований методи наївного класифікатора на цільовому тестовому наборі повідомлень з великою кількістю апострофів. Було відібрано 24 повідомлення з словами де є один і більше апострофів і протестовано виключно на цій вибірці. Результат наведено на рис. 4.

```
Correct: 15
Incorrect: 9
Accuracy: 0.625
```

Рис. 4. Ефективність Методу Байєса на прикладах з апострофами

Як бачимо, початкова ефективність оригінального методу впала з 0.86 до 0.62. Наведемо результат тестування модифікованого методу на рис. 5.

```
Correct: 19
Incorrect: 5
Accuracy: 0.7916666666666666
```

Рис. 5. Ефективність Модифікованого Методу Байєса на прикладах з апострофами

Отже, внесену зміну в метод вважатимемо ефективною, адже вона дозволила підняти ефективність методу класифікації Байєса з 0.86 до 0.91 для української мови, що є суттєвою зміною.

Висновки

В рамках даного дослідження було проаналізовано ринок засобів машинного навчання для аналізу української мови, перш за все для виконання задачі класифікації. Були використані інструменти для роботи з мовою: ВЕСУМ (Великий Електронний Словник Української мови), інструмент лематизації (знаходження кореню слова) тощо. За допомогою комбінацій інструментів розроблено модифікований метод класифікації Байєса. Оскільки готового датасету класифікованих на «спам» і «не спам» листів українською мовою знайдено не було, тому датасет був підібраний з окремих прикладів. В процесі класифікації датасету було виявлено похибку стандартного nltk Python методу класифікації, розробленого з огляду на англійську мову, який некоректно знаходив границі слів з апострофами. Для покращення аналізу тексту українською мовою розроблено модифікований метод наївної класифікації Байєса з покращеною очисткою пунктуації тексту. В результаті було отримано кращий результат відсотку правильних класифікацій – 0.91 замість 0.86. Для повноцінного аналізу потрібно випробувати цю модифікацію на значно більшому датасеті (бажано в сотні тисяч повідомлень), що стане напрямком подальших досліджень.

Таким чином, досліджена та частково вирішена проблема аналізу текстів української мови за допомогою існуючих інструментів мови програмування Python, яка полягає в словах з апострофами.

Список літератури:

[1] *Simon Kemp*. Digital 2020: 3.8 billion people use social media / Simon Kemp URL: https://wearesocial.com/blog/2020/01/digital-2020-3-8-billion-people-use-social-media (дата звернення: 25.03.2021).

[2] *Joseph Johnson*. Number of sent and received e-mails per day worldwide from 2017 to 2025 / Johnson Joseph URL: https://www.statista.com/statistics/456500/daily-number-of-e-mails-worldwide/

[3] *Барсегян А.А.* Анализ данных и процессов: учеб. пособие / А.А. Барсегян, М.С. Куприянов, И.И. Холод, М.Д. Тесс, С.И. Елизаров. – 3-е изд., перераб. и доп. – Санкт-Петербург : БХВ-Петербург, 2009. – 512 с.

[4] *Yang Y.* A re-examination of text categorization methods / Y. Yang, X. Liu // Proc. of Int.ACM Conference on Research and Development in Information Retrieval (SIGIR-99), 1999.– P. 42-49.

[5] *Вагин В.Н.* Достоверный и правдоподобный вывод в интеллектуальных системах /В. Н. Вагин, Е. Ю. Головина, А. А. Загорянская, М. В. Фомина. – Москва : Физматлит, 2004. – 704 с.

[6] *Bradley P.* Carlin Bayes and Empirical Bayes Methods for Data Analysis, Second Edition 2nd Edition / P. Bradley Carlin, A. Thomas Louis. – 2000. – 440 p.

[7] *Joachims T.* Making large-scale SVM learning practical / T. Joachims // Advances in Kernel Methods Support Vector Learning. – MIT Press, 1999. – 218 p.

[8] *Sebastiani F.* Machine learning in automated text categorization / F. Sebastiani // ACM Comput. Surv. – March 2010. – Vol. 34, No. 1. – P. 1-47.

[9] *Jean Dos Santos*. Ham or Spam? SMS Text Classification Walkthrough / Santos Dos Jean URL: https://www.kaggle.com/jeandsantos/ham-or-spam-sms-text-classification-walkthrough

[10] *Richard S.* Sutton Reinforcement Learning, second edition: An Introduction (Adaptive Computation and Machine Learning series) / Sutton S. Richard Barto G. Andrew. – 2018. – 552 p.

Надійшла до редколегії 09.04.2021