Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Центр Післядипломної освіти
(повна назва)

Кафедра Штучного інтелекту
(повна назва)

АТЕСТАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Нечіткі нейромережеві технології в задачах обробки природномовної інформації
(тема)

Виконав:
студентка 2 курсу, групи СШІмзд-18-1
Кузьміна М.О.
(прізвище, ініціали)

Спеціальність 122 – Комп'ютерні науки
(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного інтелекту (СШІ)
(повна назва спеціалізації)

Керівник к.т.н., доц. Шевченко О.Ю.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ В.О. Філатов
(підпис) (прізвище, ініціали)

2020 р.

Харківський національний університет радіоелектроніки

Центр Післядипломної освіти
(повна назва)

Кафедра Штучного інтелекту
(повна назва)

Рівень вищої освіти другий (магістерський)

Спеціальність 122 – Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного інтелекту (СШІ)
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«____»_____ 20 ___ р.

ЗАВДАННЯ

НА АТЕСТАЦІЙНУ РОБОТУ

студентові Кузьміній Марії Олександрівні
(прізвище, ім'я, по батькові)

1. Тема роботи Нечіткі нейромережеві технології в задачах обробки природномовної інформації

затверджена наказом університету від ___ _____ 20 __ р. №_____

2. Термін подання студентом роботи до екзаменаційної комісії ____ _____ 20__ р.

3. Вихідні дані до роботи Літературні джерела

4. Перелік питань, що потрібно опрацювати в роботі 1 Аналіз використання штучних нейронних мереж в завданнях обробки природомовної інформації. 2 Аналіз проблеми застосування нейронних мереж в завданнях класифікації текстової інформації. 3 Опис алгоритму і аналіз обраної нейронної мережі для класифікації текстів. 4 Реалізація програмного забезпечення.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри)_____
Рисунок 1 – Навчання з вчителем, Рисунок 2 – Навчання без вчителя, Рисунок 3 – навчання з підкріпленням, Рисунок 4 – Рекурентна мережа, Рисунок 5 – Структура гібридного нейрона «І», Рисунок 6 – Структура гібридного нейрона «АБО»., Рисунок 7 – Схема нечіткої нейронної мережі, Рисунок 8 – Посимвольний підхід, Рисунок 9 – Підхід з використанням кодування слів.

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	к.т.н., доц., Шевченко О.Ю.		

1.

2. КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання	15.01.2020	Виконано
2	Аналіз предметної галузі та постановка задачі	01.02.2020	Виконано
3	Теоретичні дослідження з предметної галузі	11.03.2020	Виконано
4	Розробка інформаційної бази	12.03.2020	Виконано
5	Розробка класифікатора	02.04.2020	Виконано
6	Підготовка пояснювальної записки	01.05.2020	Виконано
7	Підготовка презентації	20.05.2020	Виконано
8	Нормоконтроль	25.05.2020	Виконано
9	Попередній захист атестаційної роботи	26.05.2020	Виконано
10	Захист атестаційної роботи	28.05.2020	Виконано

Дата видачі завдання _____ _____ 20 __ р.

Студент _____
(підпис)

Керівник роботи _____ _____
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Записка пояснювальна: 89 с., 25 рис., 2 дод., 38 джерел.

КЛАСИФІКАЦІЯ, НЕЙРОННА МЕРЕЖА, НЕЧІТКА ЛОГІКА, НОВИННІ ТЕКСТИ, ОБРОБКА ПРИРОДНОЇ МОВИ, LVQ-МЕРЕЖА

Об'єктом дослідження є класифікація текстів новин за допомогою нейронної мережі.

Предметом дослідження є новинні тексти.

Метою дослідження є розробка інформаційно-алгоритмічного та програмного забезпечення задачі класифікації текстової інформації новинної тематики.

В процесі роботи проаналізовані проблеми застосування нейронних мереж в області обробки природної мови, розглянуті типи нейронних мереж, основні етапи класифікації, засоби, методи та підходи класифікації даних. На основі LVQ-мережі розроблено алгоритм задачі класифікації текстів. Описано інформаційне та програмне забезпечення вирішення поставленої задачі. Проведено огляд метрик оцінки якості класифікатора.

РЕФЕРАТ

Пояснительная записка: 89 с., 25 рис., 2 прил., 38 источников.

КЛАССИФИКАЦИЯ, НЕЙРОННАЯ СЕТЬ, НЕЧЕТКАЯ ЛОГИКА, НОВОСТНЫЕ ТЕКСТЫ, ОБРАБОТКА ЕСТЕСТВЕННОГО ЯЗЫКА, LVQ-СЕТЬ

Объектом исследования является классификация текстов новостей посредством нейронной сети.

Предметом исследования являются новостные тексты.

Целью исследования является разработка информационно-алгоритмического и программного обеспечения задачи классификации текстовой информации новостной тематики.

В процессе работы проанализированы проблемы применения нейронных сетей в области обработки естественного языка, рассмотрены типы нейронных сетей, основные этапы классификации, способы, методы и подходы классификации данных. На основе LVQ-сети разработан алгоритм задачи классификации текстов. Описано информационное и программное обеспечение решения поставленной задачи. Рассмотрены метрики оценки качества классификатора.

ABSTRACT

Explanatory note: 89 p., 25 fig., 2 ann., 38 sources.

CLASSIFICATION, FUZZY LOGIC, LANGUAGE PROCESSING, LVQ-NETWORK, NEURAL NETWORK, NEWS TEXTS

This degree work is devoted to news texts classification by means of neural network.

The subject of the research is the news texts.

The aim of the research is the development of information-algorithmic and program support of the task of news topic textual information classification.

In the work, problems of neural networks usage in the field of natural language processing have been analyzed; types of neural networks, main steps of classification, means, methods and approaches of data classification have been reviewed. On the basis of the LVQ-network, the algorithm of the task of texts classification has been developed. The informational and programmatic support for solving the problem have been described. Examination of the classifier quality assessment metrics has been carried out.

ЗМІСТ

ПЕРЕЛІК ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

ШНМ – штучна нейронна мережа;

ANFIS – Adaptive Network based Fuzzy Inference System – нейро-нечітка система виводу на основі адаптивної мережі;

NLP – Natural Language Processing – обробка природної мови;

TSK – Takagi-Sugeno-Kang – нечітка нейронна мережа.

ВСТУП

В сучасних умовах штучний інтелект стає способом вирішення великої кількості не тривіальних задач, які люди вирішують у повсякденному житті. Нейронна мережа, як один з передових напрямків в області штучного інтелекту, заснована на моделюванні біологічних процесів, що відбуваються в людському мозку. Будучи моделлю роботи біологічних нейронів - клітин головного мозку, штучна мережа є набором математичних інструкцій, записаних у вигляді програмного коду.

У штучній нейронній мережі кожен нейрон представлений у вигляді процесора. У ньому є канали для прийняття і виведення сигналу. На вході кожен сигнал проходить через певні з'єднання, що імітують синоптичну активність біологічних нейронів, тобто їх здатність передавати один одному інформацію. Кожен штучний нейрон може працювати тільки з однією одиницею інформації, що надходить і виконує найпростішу функцію. Однак варто їх об'єднати в мережу, як вони вже можуть справлятися з найскладнішими завданнями, які не під силу звичайним методам програмування.

На базі методів штучного інтелекту створюються і розвиваються різні програмні системи, головною особливістю яких є здатність вирішувати інтелектуальні завдання так, як це робить людина, яка розмірковує над їх вирішенням. До найбільш популярних напрямків застосування штучного інтелекту відносять прогнозування різних ситуацій, оцінку будь-якої цифрової інформації, включаючи неструктуровані дані, зі спробою надати за нею висновок, а також аналіз інформації з пошуком прихованих закономірностей (Data mining). Нейромережі оточують нас скрізь. Якщо зробити запит в пошуковій системі мережі Інтернет, нейромережі знайдуть усі відповіді. Слід наголосити, що дані обчислювальні системи здатні на набагато більше, аніж на вирішення рутинних завдань. Наприклад, є мережі, які вивчають

користувачів і пропонують рекламу відповідно до вподобань певного споживача. У той час, як деякі мережі пишуть унікальні тексти для сайтів і створюють наукові статті, інші можуть обробляти фото, як по заданим параметрам, наприклад, перетворюючи звичайний знімок в зображення за стилем схоже на зазначену автором репродукцію.

Все частіше застосовують нейромережеві архітектури і для обробки природних мов в завданнях розпізнавання мови, перекладу та відтворення мови. Постійно удосконалюється технологія перекладу іноземних слів, знову ж таки завдяки нейромережам. Незабаром з використанням перекладача, що миттєво транслює все на рідну мову, зникне потреба у володінні певною мовою.

Застосування штучного інтелекту в майбутньому буде тільки розширюватися охоплюючи все більш складні галузі знань. Безперечно нейронні мережі є помічниками. Проте, їх впровадження в деякий момент часу може призвести до знищення цілого ряду професій на ринку праці, що стане негативною стороною розвитку штучного інтелекту.

1 АНАЛІЗ ВИКОРИСТАННЯ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ В ЗАВДАННЯХ ОБРОБКИ ПРИРОДНОМОВНОЇ ІНФОРМАЦІЇ

1.1 Поняття та визначення нейромережевих технологій

Розробка штучних розумних систем, що з'єднують переваги біологічних істот і сучасної обчислювальної техніки, створює потенційні передумови для переходу до якісно нового етапу еволюції в обчислювальній техніці [1].

Нейромережевими технологіями називають комплекс інформаційних технологій, заснованих на застосуванні штучних нейронних мереж. Штучна нейронна мережа ґрунтується на сукупності з'єднаних вузлів, що є штучними нейронами (аналогічно до біологічних нейронів у головному мозку людини). Кожне з'єднання (аналогічне синапсові) між штучними нейронами може передавати сигнал від одного до іншого. Штучний нейрон, що отримує сигнал, може обробляти його і потім відправляти сигнал штучним нейронам, приєднаним до нього.

Нейрон – це обчислювальна одиниця, яка отримує інформацію, виконує над нею прості обчислення і передає її далі. Вони діляться на три основних типи: вхідний (синій), прихований (червоний) і вихідний (зелений). Також є нейрон зміщення і контекстний нейрон. У тому випадку, коли нейронна мережа складається з великої кількості нейронів, вводять термін шару. Відповідно, є вхідний шар, який отримує інформацію, n прихованих шарів (зазвичай їх не більше 3), які її обробляють і вихідний шар, який виводить результат. У кожного з нейронів є два основних параметри: вхідні дані (input data) і вихідні дані (output data). У разі вхідного нейрона: input = output. В інших, в поле input потрапляє сумарна інформація всіх нейронів з попереднього шару, після чого, вона нормалізується, за допомогою функції активації (f (x)) і потрапляє в поле output.

Синапс – це зв'язок між двома нейронами. У синапсів є 1 параметр – вага. Завдяки йому, вхідна інформація змінюється, коли передається від одного нейрона до іншого. Припустимо, є 3 нейрона, які передають інформацію наступному. Тоді є 3 ваги, відповідні кожному з цих нейронів. У того нейрона, у якого вага буде більше, та інформація і буде домінуючою в наступному нейроні (приклад – змішання квітів). Насправді, сукупність ваг нейронної мережі або матриця ваг – це своєрідний мозок всієї системи. Саме завдяки цим вагам, вхідна інформація обробляється і перетворюється в результат.

Первинною метою підходу штучної нейронної мережі було розв'язання задач таким же чином, як це робив би людський мозок. З часом увага зосередилася на відповідності певним розумовим здібностям, ведучи до відхилень від біології. Штучну нейронну мережу використовували в ряді різноманітних задач, включно з комп'ютерним зором, розпізнаванням мовлення, машинним перекладом, соціально-мережевим фільтруванням, грою в настільні та відеоігри, та медичним діагностуванням.

В основі нейромережевих технологій лежить ідея про те, що функціонування біологічного нейрона можна промоделювати відносно простим математичним моделям, а вся глибина і гнучкість людського мислення і інші найважливіші якості нервової системи визначаються не складністю нейронів, а їх великою кількістю і наявністю складної системи зв'язків між ними. У мозку людини їх число досягає 1010-1012, причому кожен з них пов'язаний з 103-104 іншими нейронами, що створює виключно комплексну структуру. Ця структура не є статичною: людина знаходиться в процесі постійного навчання, на основі інформації, що надходить в її мозок, вона набуває досвід і в результаті стає здатною вирішувати нові завдання. Накопичення досвіду виражається в зміні характеру і силі зв'язків між нейронами [2].

В поширених реалізаціях штучної нейронної мережі сигнал на з'єднанні між штучними нейронами є дійсним числом, а вихід кожного

штучного нейрону обчислюється нелінійною функцією суми його входів. Штучні нейрони та з'єднання зазвичай мають вагу, яка підлаштовується під час навчання. Вага збільшує або зменшує силу сигналу на з'єднанні. Штучні нейрони можуть мати такий поріг, що сигнал надсилається лише якщо сукупний сигнал перевищує цей поріг. Штучні нейрони зазвичай групуються в шари. Різні шари можуть виконувати різні види перетворень своїх входів. Сигнали проходять від першого (вхідного) до останнього (вихідного) шару, можливо, після проходження шарами декілька разів [2].

Компоненти нейрокомп'ютерів – нейрони і зв'язки між ними – можна комбінувати різними способами. За рахунок цього один нейрокомп'ютер можна застосовувати для вирішення різних завдань, часто не пов'язаних між собою.

Для того щоб нейронна мережа могла коректно вирішувати поставлені завдання, потрібно провести її навчання на десятках мільйонів наборів вхідних даних. Але вже розроблені різні технології прискореного навчання, сучасні відео карти дозволяють навчати нейромережі в сотні разів швидше, а нещодавно з'явилися готові, переднавченні нейромережі, зокрема, що розпізнають образи. На основі таких нейромереж можна створювати додатки, не займаючись тривалим навчанням [2].

1.2 Класифікація штучних нейронних мереж

Нейронні мережі можна класифікувати в залежності від різних якостей [3].

За характером навчання мережі поділяють на:

– навчання з вчителем, коли відомо вихідний простір рішень нейронної мережі, а також передбачається, що є вхідні сигнали і еталонні реакції на них. У процесі навчання відбувається цілеспрямована модифікація синоптичних зв'язків нейронної мережі для досягнення

максимальної відповідності умовам між реальними вихідними значеннями мережі Y і їх еталонними значеннями e (рис. 1.1) [3];

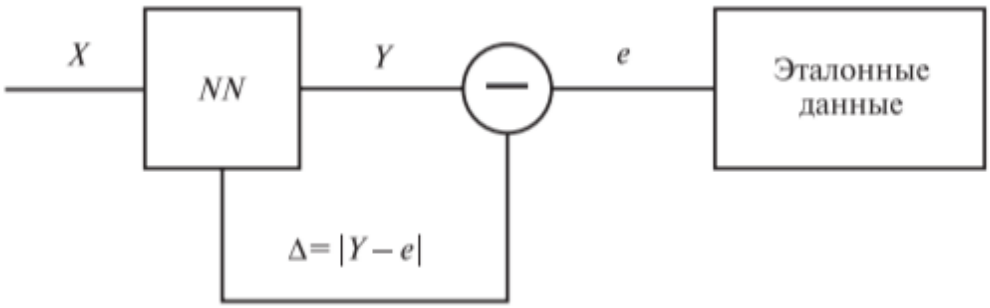


Рисунок 1.1 – Навчання з вчителем

–навчання без вчителя. В цьому випадку нейронна мережа формує вихідний простір рішень тільки на основі вхідних впливів. Такі мережі називають самоорганізованими (рис. 1.2);

Рисунок 1.2 – Навчання без вчителя

– навчання з підкріпленням (reinforcement learning). Відбувається на основі сигналу підкріплення r від зовнішнього середовища (рис. 1.3).

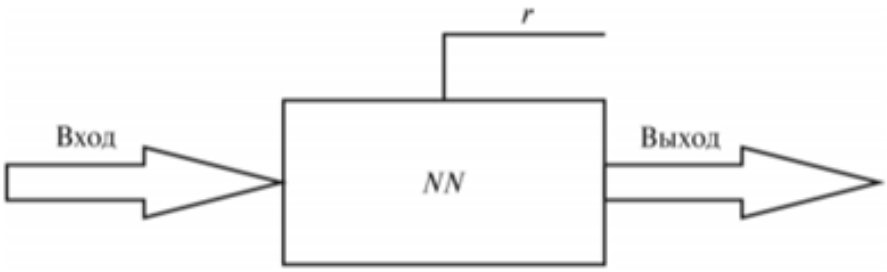


Рисунок 1.3 – Навчання з підкріпленням

За характером налаштування синапсів поділяють на:

– мережі з фіксованими зв'язками. В цьому випадку вагові коефіцієнти нейронної мережі обираються відразу, згідно з умовою задачі:

, (1.1)

де W – вагові коефіцієнти мережі.

– мережі з динамічними зв'язками. Для них в процесі навчання відбувається налаштування синоптичних зв'язків:

(1.2)

За архітектурою і навчанням поділяють на:

– персептронні нейронні мережі. Вони характеризуються, зазвичай, навчанням з вчителем. Архітектура їх базується на багатошаровому персептроні, а в основі навчання лежить метод градієнтного спуску. До них належать багатошарові персептрони та рекурентні нейронні мережі, в яких присутній зворотний зв'язок між входом та виходом. Вихідне значення при цьому визначається в залежності як від вхідних, так і попередніх вихідних значень нейронної мережі (рис. 1.4);

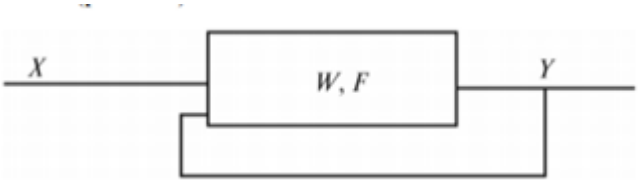


Рисунок 1.4 – Рекурентна мережа

До вищезазначеного також відносять згорткові нейронні мережі, що становлять подальший розвиток персептрона і неокогнітрона для обробки зображень та глибинні нейронні мережі, що здійснюють глибоке нелінійне ієрархічне перетворення інформації.

– самоорганізовані нейронні мережі, навчання без вчителя. Таке навчання базується тільки на сигналах від зовнішнього середовища. До них відносять нейронні мережі Кохонена та нейронні мережі адаптивного резонансу;

– релаксаційні нейронні мережі, в яких циркуляція інформації відбувається до тих пір, поки не перестануть змінюватися вихідні значення нейронної мережі (стан рівноваги). До них відносять нейронні мережі Хопфілда, нейронні мережі Хеммінга та двунаправленну асоціативну пам'ять;

– гібридні нейронні мережі. Відрізняються застосуванням двох підходів до навчання - з вчителем і без вчителя. До них відносять нейронні мережі зустрічного розповсюдження (counter propagation networks), нейронні мережі з радіально-базисною функцією активації (RBF networks) та нечіткі нейронні мережі, що характеризуються застосуванням нечіткої логіки і нейронних мереж.

– нейронні імунні мережі, в яких застосовуються штучні імунні системи і нейронні мережі [3].

1.3 Нечіткі нейронні мережі

1.3.1 Нечіткі відношення та їх властивості

Нечітка логіка виведена з теорії нечітких множин, що має справу з міркуваннями, які більшою мірою є наближеними, аніж точним. Істинність в нечіткій логіці відображає належність до нечітко визначених множин. В нечіткій логіці рішення можуть бути прийняті на основі неточно визначених, але, тим менш, дуже важливих характеристик. Предметом нечіткої логіки вважається дослідження міркувань в умовах нечіткості, розмитості, схожих з міркуваннями в звичному сенсі, і їх застосування в обчислювальних системах [4].

Символічна нечітка логіка ґрунтується на понятті t-норми. Після вибору деякої t-норми (її можна ввести кількома різними способами) з'являється можливість визначити основні операції над пропозиціональними змінними: кон'юнкцію, диз'юнкцію, імплікацію, заперечення та інші. Неважко довести теорему про те, що дистрибутивність, присутня в класичній логіці, виконується тільки в разі, коли в якості t-норми обирається t-норма Геделя. Крім того, в силу певних факторів, в якості імплікації найчастіше обирають операцію residium (вона також залежить від вибору t-норми). Визначення основних операцій, зазначених вище, призводить до формального визначення базисної нечіткої логіки, яка має багато спільного з класичною булевозначною логікою (з обчисленням висловлювань). Існують три основні базисні нечіткі логіки: логіка Лукасевича, логіка Геделя і ймовірнісна логіка. Цікаво, що об'єднання будь-яких двох з трьох перерахованих вище логік призводить до класичної булевозначної логіки.

Основне поняття нечіткої логіки в широкому сенсі – нечітка множина, яка визначається за допомогою узагальненого поняття характеристичної функції. Потім вводяться поняття об'єднання, перетину і доповнення множин (через характеристичну функцію), поняття нечіткого

відношення, а також одне з найважливіших понять – поняття лінгвістичної змінної. Оскільки нечіткі множини описуються функціями приналежності, а t-норми і k-норми звичайними математичними операціями, можна уявити нечіткі логічні міркування у вигляді нейронної мережі. Для цього функції приналежності треба інтерпретувати як функції активації нейронів, передачу сигналів як зв'язки, а логічні t-норми і k-норми, як спеціальні види нейронів, що виконують відповідні математичні операції. Існує велика різноманітність подібних нейро-нечітких мереж (neuro-fuzzy network). Наприклад, ANFIS – адаптивна нейро-нечітка система виведення [4].

В нечітких системах найчастіше областю визначення нечітких множин є багатовимірний простір. При цьому особливий інтерес представляють множини з двовимірною областю визначення. Нечіткі множини на багатовимірних областях визначення задаються як нечіткі відношення. Нечітким k-арним відношенням, заданим на множині (універсумі) X_1, X_2, ..., X_k, називається нечітка множина R, визначена на декартовому добутку $X_1 \times X_2 \times ... \times X_k$. Відношення (чітке або нечітке), побудоване на основі двох множин, називається бінарним, на основі трьох множин – тернарним, на основі k множин – k-арним. Відношення може бути задано на одному універсумі.

Нечіткі відношення можуть бути задані різними способами:

– у формі списку з явним перерахуванням всіх кортежей відношення і відповідних їм функцій приналежності. Цей спосіб застосовується для відношень з невеликим числом кортежів;

– аналітично у формі виразу для функції приналежності;

Для бінарних відношень додатково можуть бути використані наступні способи:

– графічно у вигляді деякої поверхні. При цьому незалежними змінними є значення універсумів X_1 і X_2, а третя координата є відповідним значенням функції приналежності;

– у вигляді матриці кінцевого бінарного відношення, рядки якої відповідають першим елементам кортежів, а стовпці – другим елементам кортежів. Елементами матриці є відповідні значення функції приналежності;

– у вигляді нечіткого графа, вершини якого відповідають елементам універсумів X_1 і X_2, а дуга, що з'єднує вершини $x_i \in X$ і $x_j \in X$, показує, що кортеж, (x_i, x_j) входить в відношення. Біля кожної дуги вказується функція приналежності відповідного кортежу.

Так як нечітке відношення є нечіткою множиною, то зберігаються визначення над відношеннями. Операції над нечіткими відношеннями можуть бути визначені за допомогою t-норм і s-норм. Найчастіше потрібно виконати операції над нечіткими множинами, які задані на універсумі X_1 і X_2. Такі множини за допомогою операції циліндричного продовження (розширення) заздалегідь приводять до нечітких відношень, заданих на декартовому добутку X_1 і X_2, а потім виконують операцію над відношеннями, заданому на одному універсумі. Циліндричне продовження визначається наступним чином. Нехай X_1 і X_2 – чіткі множини (функції приналежності елементів яких дорівнюють одиниці), а A – нечітка множина, задана на X_1. Циліндричним продовженням A* множини A на область визначення $X_1 \times X_2$ називається відношення, що є декартовим добутком $A \times X_2$. У матричному поданні відношення A* стовпці матриці функцій приналежності будуть однаковими і рівними значенням функції приналежності нечіткої множини A*.

За допомогою прийняття рішень в ШНМ, заснованої на нечіткій логіці, можна створити потужну систему управління. Вочевидь, ці дві концепції добре працюють разом: алгоритм логічного виводу з трьома нечіткими станами (наприклад, холодний, теплий, гарячий) міг би бути реалізований в апаратному вигляді при використанні істиннісних значень (0.8, 0.2, 0.0) в якості вхідних значень для трьох нейронів, кожен з яких становить одну з трьох множин [5].

Кожен нейрон обробляє вхідну величину відповідно до своєї функції і отримує вихідне значення, яке далі буде вхідним значенням для другого шару нейронів. Наприклад, нейрокомп'ютер для обробки зображень може зняти численні обмеження з відеозапису, висвітлення і налаштувань апаратури. Такий ступінь свободи стає можливим завдяки тому, що нейронна мережа дозволяє побудувати механізм розпізнавання за допомогою вивчення прикладів. В результаті система може бути навчена розпізнаванню придатних і дефектних виробів при сильному і слабкому освітленні, при їх розташуванні під різними кутами. Механізм логічного виводу починає працювати з «оцінки» умов освітлення (іншими словами, встановлює ступінь подібності з іншими умовами освітлення, при яких система знає, як діяти). Після цього система виносить рішення щодо утримання зображення використовуючи критерії, що засновані на даних умовах освітлення. Оскільки система розглядає умови освітлення як нечіткі поняття, механізм логічного виводу легко визначає нові умови за відомими прикладами. Чим більше прикладів вивчає система, тим більший досвід набуває механізм обробки зображень. Цей процес навчання може бути досить легко автоматизований, наприклад, за рахунок попереднього сортування за групами деталей з близькими властивостями для навчання за областями подібностей і відмінностей. Ці спостережувані подібності та відмінності можуть далі надавати інформацію ШНМ, завдання якої полягає в сортуванні деталей, що надходять за цими категоріями [5].

Нейрокомп'ютер для обробки зображень підходить для додатків, де діагностика спирається на досвід і експертну оцінку оператора, а не на моделі і алгоритми. Процесор може побудувати механізм розпізнавання з простих коментарів зображення, зроблених оператором, після чого вилучити характеристики або вектори ознак з об'єктів, забезпечених коментарями, і передати їх в нейронну мережу. Вектори ознак, що описують видимі об'єкти, можуть бути такими простими як значення рядка пікселів, гістограма або розподіл інтенсивності, профілі розподілу

інтенсивності або градієнти за відповідними осями. Більш складні ознаки можуть включати елементи вейвлет-перетворення і швидкого перетворення Фур'є. Після навчання на прикладах нейронна мережа здатна до узагальнення і може класифікувати ситуації, що ніколи раніше спостерігалися, пов'язуючи їх зі схожими ситуаціям з прикладів. З іншого боку, якщо система схильна до зайвої свободи і узагальненню ситуацій, її поведінку в будь-який час може бути скориговано за рахунок навчання протилежним прикладам. З точки зору нейронної мережі ця операція полягає в зменшенні областей впливу існуючих нейронів для узгодження з новими прикладами, які знаходяться в суперечності з існуючим відображенням простору рішень [5].

Важливим фактором, що визначає визнання ШНМ, є самостійне і адаптивне навчання. Це означає, що пристрій має володіти здатністю вивчати об'єкт з мінімальним втручанням оператора або взагалі без його втручання.

1.3.2 Основні поняття і визначення нечітких нейронних мереж

Основним компонентом в процедурах нечіткого виводу є база правил нечітких продукцій. У той же час існують цілі класи прикладних задач, в яких виявлення і побудова правил нечітких продукцій неможливі або пов'язані з серйозними труднощами концептуального характеру. До таких завдань відносяться завдання розпізнавання образів, екстраполяції і інтерполяції функціональних залежностей, класифікації та прогнозування, нелінійного і ситуаційного управління, а також інтелектуального аналізу даних (Data Mining) [6].

Загальною особливістю подібних завдань є існування деякої залежності або відношення, що зв'язує вхідні і вихідні змінні моделі системи, що подається у формі так званого «чорного ящика». При цьому виявлення і визначення цієї залежності в явному теоретико-множинному

або аналітичному вигляді не є можливим або через недолік інформації про проблемну область, що моделюється або складність обліку різноманіття чинників, що впливають на характер цього взаємозв'язку. Для конструктивного вирішення подібних завдань розроблено спеціальний математичний апарат, який отримав назву нейронних мереж.

Перевагою моделей, побудованих на основі нейронних мереж, є можливість отримання нової інформації про проблемну область в формі деякого прогнозу. При цьому побудова та налаштування нейронних мереж здійснюється за допомогою їх навчання на основі наявної і доступної інформації.

Недоліком нейронних мереж є уявлення знань про проблемну область у спеціальному вигляді, яке може істотно відрізнятися від можливої змістовної інтерпретації існуючих взаємозв'язків і відношень.

Нечітка нейронна (гібридна) мережа – це нейронна мережа з чіткими сигналами, вагами і активаційною функцією, але з об'єднанням x_i і w_i, p_1 і p_2 з використанням t-норми, t-конорми або деяких інших безперервних операцій. Входи, виходи і ваги нечіткої нейронної мережі дійсні числа, що належать відрізку [0, 1]. Нечіткою нейронною мережею зазвичай називають чітку нейронну мережу, яка побудована на основі багатошарової архітектури з використанням «І», «АБО» нейронів. Нечіткий нейрон «І». Сигнали x_i і w_i в даному випадку об'єднуються за допомогою трикутної конорми [6]:

, (1.3)

а вихід утворюється із застосуванням трикутної норми (рис. 1.5).

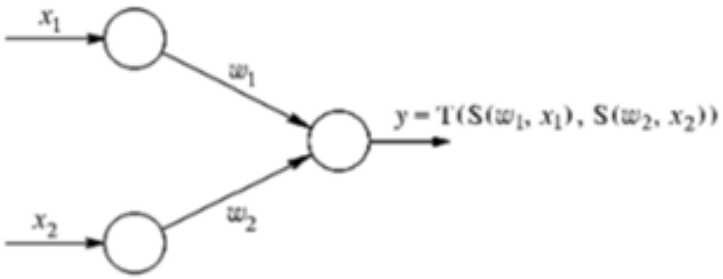


Рисунок 1.5 – Структура гібридного нейрона «І»

Нечіткий нейрон «АБО». Сигнали x_i і w_i тут об'єднуються за допомогою трикутної норми, а вихід утворюється із застосуванням трикутної конорми (рис. 1.6).

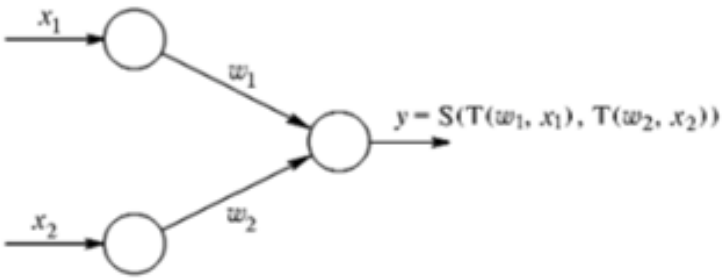


Рисунок 1.6 – Структура гібридного нейрона «АБО»

Нечітка нейронна мережа як правило складається з чотирьох шарів: шару фазифікація вхідних змінних, шару агрегування значень активації умови, шару агрегування нечітких правил і вихідного шару.

Нечітка нейронна мережа є набором нечітких правил, що описують класи в наявному наборі вихідних даних, і нечітку систему виводу для їх переробки з метою отримання результату діагностики. Фрагмент нечіткої нейронної мережі наведено на рисунку 1.7 [6].

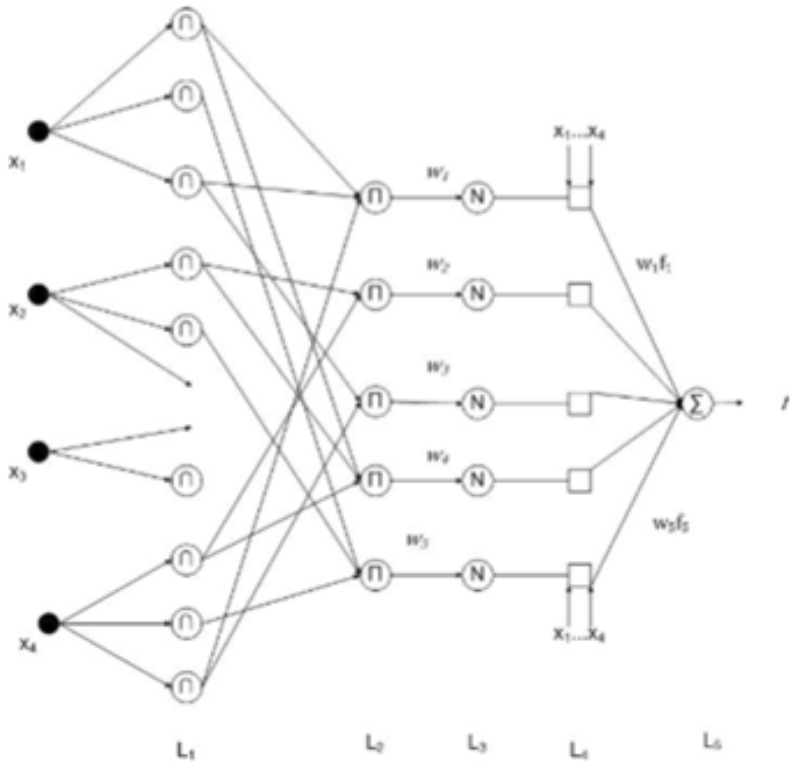


Рисунок 1.7 – Схема нечіткої нейронної мережі

На схемі показана нечітка нейронна мережа з чотирма входами (n = 4). Шари позначені символами від L_1 до L_5. Елементи, позначені символом П (мультиплікатори), перемножують всі вхідні сигнали, елементи, позначені символом Σ (суматори) – підсумовують їх. Призначення шарів, наступне: перший шар – терми вхідних змінних; другий шар – антецеденти (посилки) нечітких правил; третій шар – нормалізація ступенів виконання правил; четвертий шар – укладення правил; п'ятий шар – агрегування результату, отриманого за різними правилами. Входи мережі в окремий шар не виділяються.

Кожен елемент шару 1 (L_1) становить один терм з функцією приналежності. Входи мережі з'єднані тільки зі своїми термами. Кількість вузлів першого шару дорівнює сумі потужностей терм-множин вхідних змінних.

Кількість вузлів шару 2 (L_2) дорівнює m. Кожен вузол цього шару відповідає одному нечіткому правилу. Вузол другого шару з'єднаний з тими вузлами першого шару, які формують антецеденти відповідного правила. Отже, кожен вузол другого шару може приймати від 1 до n вхідних сигналів. Виходом вузла є ступінь виконання правила (вага деякого правила). Кількість елементів цього шару дорівнює кількості правил N. Кожен вузол пов'язаний з попереднім шаром таким чином, що вузол шару L_2, відповідний k-м правилом, з'єднаний з усіма вузлами шару L1, що відповідають нечітким множинам суджень цього правила.

На відміну від «чистих» нейронних мереж, кожен шар в цілому і окремі складові його елементи, також, як і конфігурація зв'язків, всі параметри і ваги мають фізичну інтерпретацію. Ця властивість виявляється надзвичайно важливою, оскільки знання не розподіляються по мережі і можуть бути легко локалізовані і при необхідності відкориговані експертом-спостерігачем.

Розглянуті вище етапи нечіткого виводу можуть бути реалізовані неоднозначним чином, оскільки містять окремі параметри, які повинні бути фіксовані або специфіковані. Тим самим вибір конкретних варіантів параметрів кожного з етапів визначає певний алгоритм, який реалізує нечіткий вивід в системах правил нечітких продукцій. Один з таких алгоритмів – алгоритм Сугено. Його основна особливість полягає в тому, що укладення правил задаються лінійною функцією від входів. Швидкі алгоритми навчання та інтерпретованість накопичених знань – ці фактори зробили сьогодні нечіткі нейронні мережі одним з найбільш перспективних і ефективних інструментів м'яких обчислень.

Найбільшого поширення в даний час отримали архітектури нечіткої нейронної мережі виду ANFIS і TSK. Доведено, що такі мережі є універсальними аппроксиматорами [7].

1.3.3 Навчання нечіткої нейронної мережі

В існуючих системах з нечіткими нейронними мережами одним з найважливіших питань є розробка оптимального методу налаштування нечіткої бази правил, на основі навчальної вибірки, для отримання конструктивної і оптимальної моделі нечіткої нейронної мережі. В основному нечіткі правила описуються експертами або операторами згідно з їх знаннями і досвідом про відповідні процеси. Але в разі розробки нечіткої нейронної мережі досить важко або майже неможливо відразу отримати нечіткі правила або функції приналежності внаслідок неясності, неповноти або складності системи, що описується. У таких випадках доцільним вважається уточнення нечітких правил і функцій приналежності, використовуючи спеціальні алгоритми навчання [7].

Так як нечітка мережа ANFIS, подається у вигляді багатошарової структури з прямим розповсюдженням сигналу, а значення вихідної змінної можна змінювати, коригуючи параметри елементів верств, то для навчання цієї мережі можна використовувати градієнтний алгоритм.

Розглянемо навчання вищезазначених мереж виду ANFIS і TSK.

Так як нечітка мережа ANFIS, описана раніше, подається у вигляді багатошарової структури з прямим розповсюдженням сигналу, а значення вихідної змінної можна змінювати, корегуючи параметри елементів шарів, то для навчання цієї мережі можна використовувати градієнтний алгоритм.

Основною характерною рисою даного підходу є те, що налаштування параметрів функцій приналежності здійснюється без модифікації бази правил.

На першому етапі навчання для кожного прикладу з навчальної вибірки за значеннями вхідних змінних x_1 (k), x_2 (k)...., k_k нечітка мережа розраховує значення вихідної змінної y (k).

На другому етапі обчислюється функція помилки для всіх прикладів навчальної вибірки:

. (1.4)

На третьому етапі коригуються значення (a_{ij}, b_{ij}) за кожним прикладом навчальної вибірки, на основі співвідношень. Етапи 1 – 3 ітераційно повторюються, і процедура коригування значень всіх параметрів вважається завершеною у тому випадку, коли значення функції помилки за кожним прикладом навчальної вибірки не перевищує деякого встановленого порога:

. (1.5)

У цьому випадку вважається, що нечітка мережа успішно навчилася. При навчанні за допомогою алгоритму зворотнього поширення помилки забезпечується властивість єдиності подання лінгвістичних термів за рахунок зв'язаної адаптації параметрів функцій приналежності [7].

Варто також розглянути гібридний алгоритм навчання. Даний алгоритм є алгоритмом навчання нечіткої мережі TSK. Нечіткі правила даної моделі мають вид, що відображається формулою:

. (1.6)

У гібридному алгоритмі параметри, що підлягають адаптації, діляться на 2 групи. Перша з них складається з лінійних параметрів p_{kj} третього шару (ваг), а друга група – з параметрів нелінійної функції приналежності першого шару – це параметри pk_0.

У гібридному алгоритмі параметри, що підлягають адаптації, діляться на 2 групи. Перша з них складається з лінійних параметрів p_{kj} третього шару (ваг), а друга група – з параметрів нелінійної функції приналежності першого шару - це параметри pk_0. Уточнення параметрів відбувається в 2 етапи. На першому етапі при фіксації окремих значень параметрів функції приналежності (в першому циклі – це значення, які

отримані шляхом ініціалізації), розв'язуючи систему лінійних рівнянь, обчислюються лінійні параметри p_{kj}. При відомих значеннях функції приналежності залежність для виходу можна представити у вигляді лінійної форми щодо параметра p_{kj}. При практичній реалізації гібридного методу навчання нечітких мереж домінуючим фактором їх адаптації вважається перший етап, на якому ваги p_{kj} підбираються з використанням псевдоінверсії за один крок. Для врівноваження його впливу другий етап багато разів повторюється в кожному циклі.

З використанням методу оптимізації (метод найменших квадратів) оцінюються коефіцієнти висновків правил, так як вони лінійно пов'язані з виходом мережі. Кожна ітерація процедури налаштування виконується в два етапи. На першому етапі на входи подається навчальна вибірка, і за невязкою між бажаним і дійсною поведінкою мережі ітераційним методом найменших квадратів знаходяться оптимальні параметри вузлів четвертого шару. На другому етапі залишкова невязка передається з виходу мережі на входи, і методом зворотного поширення помилки модифікуються параметри вузлів першого шару. При цьому знайдені на першому етапі коефіцієнти виводів правил не змінюються. Ітераційна процедура налаштування триває доки невязка перевищує заздалегідь встановлене значення. Щоб визначити опції приналежностей крім методу зворотного поширення помилки можуть використовуватися і інші алгоритми оптимізації [8].

1.4 Нейромережеві методи в обробці природної мови

Обробка природної мови – це наука про проектування методів і алгоритмів, які приймають або породжують неструктуровані дані природної мови. Розуміння і породження мови за допомогою комп'ютерів - надзвичайно важке завдання. Кращі методи роботи з мовними даними ґрунтуються на алгоритмах машинного навчання з вчителем, які

намагаються вивести патерни і закономірності використання з множини попередньо анотованих пар вхідних і вихідних текстів. Розглянемо, наприклад, задачу класифікації документа за однією з чотирьох категорій: Спорт, Політика, Світська хроніка та Економіка. Очевидно, що слова, які містяться в документі дають цілком певні вказівки, але яку саме вказівку дає дане слово? Виписати відповідні правила досить важко. Однак читачі легко можуть віднести документ до теми, а потім, спираючись на декілька сотень анотованих таким чином документів в кожній категорії, алгоритм машинного навчання з вчителем може вивести патерни використання слів, які допоможуть класифікувати нові документи. Методи машинного навчання відмінно працюють в задачах, де визначити хороший набір правил дуже важко, а анотувати вхідні приклади вихідними мітками порівняно просто.

Крім проблем, пов'язаних з обробкою неоднозначних і варіативних вхідних даних в системі з погано визначеними і відсутніми наборами правил, у природної мови є і додаткові властивості, які ще більше ускладнюють розробку обчислювальних підходів на основі машинного навчання: дискретність, композиційність і розрідженість.

Мова за своєю природою символічна і дискретна. Основними елементами писемної мови є літери. Літери утворюють слова, що позначають предмети, поняття, події, дії і ідеї. Мова має властивість композиційних: літери утворюють слова, слова утворюють фрази і речення. Сенс фрази може бути більше сенсу складових її слів і визначається набором заплутаних правил. Тому, щоб інтерпретувати текст, доводиться піднятися над рівнем букв і слів і розглядати довгі послідовності слів, наприклад речення або навіть повні документи. Поєднання описаних вище властивостей веде до розрідженості даних. Число комбінацій слів (дискретних символів), що мають сенс, практично нескінченно. Число допустимих речень величезна, немає ніякої надії перерахувати їх усі [9].

Нейронні мережі надають ефективний механізм навчання, надзвичайно привабливий для використання в задачах обробки природної мови. Головний компонент мовної нейронної мережі – шар занурення, тобто відображення дискретних символів на безперервні вектори в просторі порівняно невеликої розмірності. В результаті занурення слова перетворюються з ізольованих дискретних символів в математичні об'єкти, над якими можна виконувати різні дії. Зокрема, якщо за міру відстані між словами взяти відстань між векторами, то буде простіше узагальнити вплив одного слова на інше. Таке уявлення слів векторами мережа знаходить в процесі навчання. Піднімаючись вгору по ієрархії, мережа також навчається комбінувати вектори слів способами, корисними для прогнозування. Ця можливість в деякій мірі компенсує дискретність і розрідженість даних. Існує два основних види архітектури нейронних мереж, які можна по-різному комбінувати: мережі прямого поширення і рекурентні / рекурсивні мережі. Мережі прямого поширення, між іншим багатошарові персептрони, дозволяють працювати з вхідними даними фіксованого розміру або з даними змінного розміру, якщо можна не звертати уваги на порядок елементів. Якщо завантажити в мережу безліч вхідних компонентів, то вона навчиться комбінувати їх осмисленими способами. Багатошарові персептрони можна використовувати в тих випадках, де раніше застосовувалася лінійна модель. Не лінійність мережі, а також можливість інтегрувати в неї раніше навчені занурення слів часто призводять до видатної точності класифікації.

Згорткові нейронні мережі – це спеціалізовані архітектури, що відрізняються здатністю виділяти локальні патерни в даних: на вхід їм подаються дані довільного розміру, а вони виділяють осмислені локальні патерни, чутливі до порядку слів, незалежно від того, в якому місці вхідних даних вони зустрічаються. Вони дуже добре справляються з ідентифікацією фраз в дійсному стані і ідіом заздалегідь обмеженої довжини в довгих реченнях або документах. Алфавіт – це упорядкований

набір символів. Нехай цей алфавіт складається з m символів. Кожен символ алфавіту в тексті закодований з допомогою 1 – m кодування. У випадку, коли в тексті зустрінеться символ, який не увійшов до алфавіту, то його необхідно закодувати вектором довжини m, що складається з одних нулів. Далі, з тексту слід обрати перші I символів. Параметр I повинен бути більшим, 18 щоб в перших I символах містилося достатньо інформації для визначення класу всього тексту (рис. 1.8) [10].

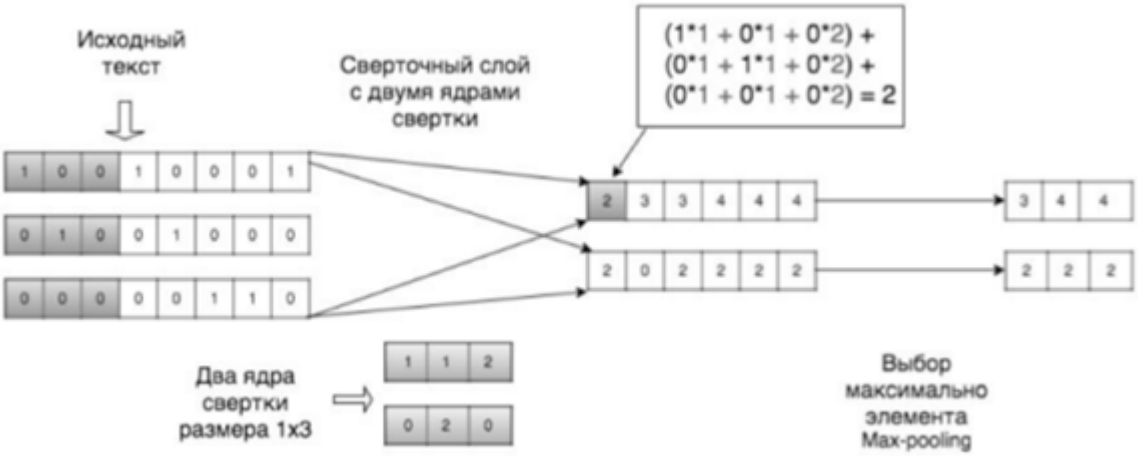


Рисунок 1.8 – Посимвольний підхід

Потім вихідні вектори об'єднуються в матрицю розміру m × l, в якій в кожен стовпець матиме не більше однієї одиниці. Кожен рядок отриманої матриці використовується як окрема карта ознак. На вхід згорткової нейронної мережі подається m карт ознак розміру 1 × l аналогічно зображенню. Архітектуру мережі необхідно вибирати виходячи з завдання. На рисунку 1.9 наведено приклад посимвольного підходу для I = 6, m = 3.

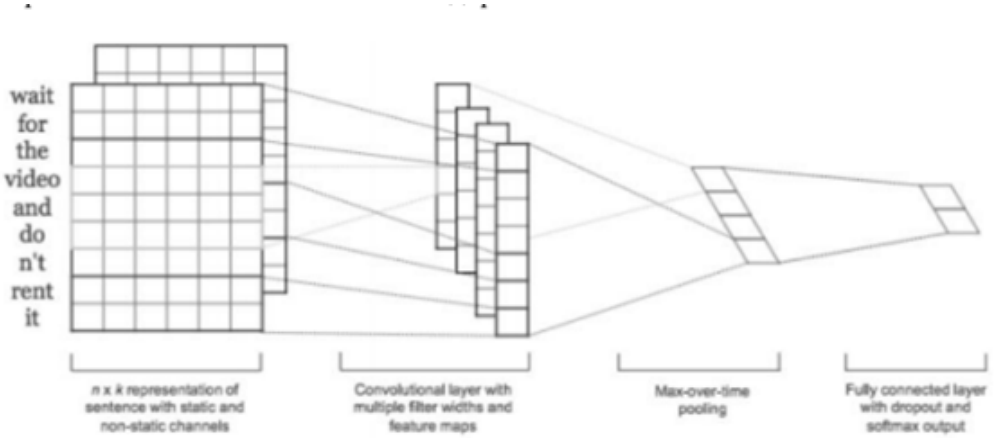


Рисунок 1.9 – Підхід з використанням кодування слів

В даному підході кожному слову в тексті зіставляється вектор фіксованої довжини, потім з отриманих векторів для кожного об'єкта вибірки складається матриця, яка аналогічно зображень подається на вхід згортковій нейронній мережі.

Рекурентні нейронні мережі – це спеціалізовані моделі для послідовних даних. Вони приймають вхідну послідовність об'єктів і породжують вектор фіксованої довжини, який підсумовує її. Сенс слів «підсумувати послідовність» залежить від завдання (наприклад, інформація, необхідна для відповіді на питання про емоційне забарвлення речення, відрізняється від тієї, що необхідна для відповіді на питання про його граматичну правильність). Тому рекурентні мережі рідко використовуються самі по собі, а їх цінність полягає в тому, що це компоненти, що допускають навчання, які можна подати на вхід іншим компонентам мережі та навчити спільній роботі. Наприклад, вихід рекурентної мережі можна подати на вхід мережі прямого поширення, яка спробує передбачити деяке значення. Рекурентні мережі є досить виразними моделями для послідовностей і є, мабуть, найкориснішими, що можуть запропонувати нейронні мережі обробці мов. Вони дозволяють відмовитися від марковского припущення, що переважає в NLP протягом декількох десятиліть, і проектувати моделі, в яких умовами можуть бути

цілі речення. При цьому вони можуть при необхідності враховувати порядок слів і не схильні до проблем статистичного оцінювання, що виникає з розрідженістю даних. Ця можливість дає помітний виграш в мовному моделюванні – завданні про пророкування ймовірності наступного слова в послідовності (або, що те ж саме, ймовірності речення), – яке є наріжним каменем багатьох додатків NLP. Рекурсивні мережі узагальнюють рекурентні мережі з послідовностей на дерева [11].

Багато задач в природній мові структуровані, тобто потребують породження складних вихідних структур типу послідовностей або дерев. Нейромережеві моделі придатні і для цих цілей - або шляхом адаптації відомих алгоритмів структурного прогнозування для лінійних моделей, або завдяки використанню нових архітектур, таких як моделі послідовностей (кодировщик-декодер), які називають також моделями умовної генерації (conditioned generation model). Деякі мовні задачі прогнозування пов'язані один з одним в тому сенсі, що знання про те, як вирішити одну з них, допомагає при навчанні іншим. Крім того, якщо анотованих вчителем навчальних прикладів може не вистачати, то вже нестачі в початкових (неанотованих) текстових даних достеменно не спостерігається. Чи можна навчатися на основі неанотованих даних або результатів навчання споріднених завдань? Нейронні мережі пропонують досить цікаві можливості як для багатозадачного навчання (multitask learning), тобто навчання на основі результатів для споріднених завдань, так і для навчання з частковим залученням вчителя (навчання на зовнішніх неанотованих даних) [12].

1.5 Постановка задач дослідження

Технології обробки природної мови сьогодні зробили крок далеко вперед, і чимала заслуга в цьому належить машинному навчанню, що застосовується, зокрема, для розуміння текстів. Існує велика кількість

методів обробки природної мови, проте більшість методів успішно використовують тільки уявлення слів, ігноруючи синтаксис і семантику, які можна вивести з синтаксичної структури речень. Нейронні мережі надають можливість іншого підходу до роботи з реченнями.

Метою дипломної роботи є дослідження використання нечітких нейромережевих технологій в завданнях обробки природної мови.

Об’єктом дослідження є нечітка нейронна мережа.

Предметом дослідження є застосування нечітких нейромережевих технологій для обробки природномовної інформації.

Досягнення мети дипломної роботи обумовило необхідність вирішення таких завдань:

–визначення поняття нейромережевих технологій;

–аналіз існуючих видів нейронних мереж та їх властивостей;

–вивчення основних понять в області нечіткої логіки, що використовується в нечітких нейронних мережах для розуміння процесу навчання такої мережі;

–аналіз використання нейромережевих методів в обробці природної мови.

Інформаційну базу дослідження становлять підручники, наукові розробки та публікації провідних вітчизняних і зарубіжних вчених у сфері використання нейронних мереж в області NLP.

2 АНАЛІЗ ПРОБЛЕМИ ЗАСТОСУВАННЯ НЕЙРОННИХ МЕРЕЖ В ЗАВДАННЯХ КЛАСИФІКАЦІЇ ТЕКСТОВОЇ ІНФОРМАЦІЇ

2.1 Актуальність завдання класифікації текстів. Основні етапи класифікації

Вирішення задачі класифікації є одним з найважливіших застосувань нейронних мереж. Класифікація становить собою завдання віднесення зразка до однієї з декількох попарно не пересічних множин. ЇЇ актуальність обумовлена постійно зростаючим обсягом інформації в інтернеті і потребою в ній орієнтуватися. Класифікація знаходить застосування у наступних задачах:

– боротьба зі спамом. Спам – це небажані розсилки, які можуть надходити на адресу електронної пошти. Вони можуть містити рекламні пропозиції або комп'ютерні віруси. Завдання боротьби зі спамом полягає в класифікації всіх листів на два класи: спам і не спам;

– розпізнавання емоційного забарвлення текстів. Завдання полягає в оцінки думку автора по відношенню до об'єктів, наприклад на основі відгуків про ці об'єкти. Часто таке завдання необхідно вирішувати для видачі релевантних рекомендацій;

– поділ сайтів за тематичними каталогами. Дане завдання вирішується пошуковими системами і передбачає обробку документів і віднесення їх до однієї з декількох категорій, перелік яких заздалегідь заданий;

– персоніфікація реклами. Контекстна реклама є основним джерелом доходу IT компаній. Вона відображається відвідувачам інтернет-сторінки, сфера інтересів яких потенційно збігається або перетинається з тематикою товару або послуги, що рекламується, цільової аудиторії, що підвищує ймовірність їх відгуку на рекламу. Сфера інтересів визначається за текстом інтернет-сторінок переглянутих користувачем [13].

Формально задачу класифікації текстових документів описують набором множин. Множина документів подається у вигляді:

. (2.1)

Категорії документів подаються множиною:

. (2.2)

У задачі класифікації потрібно на основі цих даних побудувати процедуру, яка полягає в знаходженні найбільш імовірної категорії з множини C для досліджуваного документа d_i. Більшість методів класифікації текстів так чи інакше засновані на припущенні, що документи, що відносяться до однієї категорії, містять однакові ознаки (слова або словосполучення), і наявність або відсутність таких ознак в документі говорить про його належність чи неналежність до тієї чи іншої теми. Таким чином, для кожної категорії має бути множина ознак:

,

. (2.3)

Таку множину ознак найчастіше називають словником, так як вона складається з лексем, які включають слова і / або словосполучення, що характеризують категорію. Подібно категоріям кожен документ також має ознаки, за якими його можна віднести з деякою мірою вірогідності до однієї або декількох категорій:

. (2.4)

Множина ознак всіх документів має збігатися з множиною ознак категорій, тобто:

. (2.5)

Необхідно зауважити, що дані набори ознак є відмінною рисою класифікації текстових документів від класифікації об'єктів в Data Mining, які характеризуються набором атрибутів. Рішення про віднесення документа d_i до категорії c_r приймається на підставі перетину:

. (2.6)

Завдання методів класифікації полягає в тому, щоб найкращим чином обрати такі ознаки і сформулювати правила, на основі яких буде прийматися рішення про віднесення документа до категорії [14].

Класифікація текстів складається з етапу попередньої обробки текстів, перекладу текстів в дійсний простір ознак, де кожному документу буде зіставлено вектор фіксованої довжини, вибір алгоритму машинного навчання для класифікації. На рисунку 2.1 надана схема, що описує вищезазначені етапи класифікації текстів.

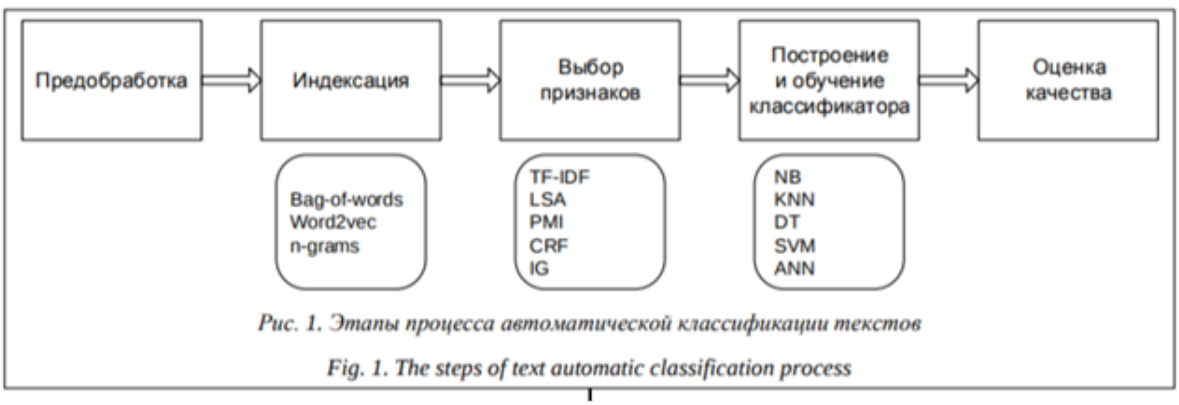


Рисунок 2.1 – Схема етапів класифікації текстів

Всі тексти на природній мові мають велику кількість слів, які не несуть інформації щодо даного тексту. Наприклад, в англійській мові такими словами є артиклі, в російській до них можна віднести прийменники, сполучники, частки. Дані слова називають шумовими або стоп-словами. Для досягнення кращої якості класифікації на першому етапі попередньої обробки текстів зазвичай необхідно видаляти такі слова. Другий етап попередньої обробки текстів – приведення кожного слова до основи, однакової для всіх його граматичних форм. Це необхідно, так як слова, що несуть один і той же сенс можуть бути записані в різній формі. Наприклад, одне й те ж слово може зустрітися в різних відмінах, мати різні приставки та закінчення. Попередня обробка тексту містить токенізацію, видалення функціональних слів (семантично нейтральних слів, таких як союзи, прийменники, артиклі та інше). Далі здійснюється морфологічний аналіз (здійснюється розмітка за частинами мови і стемматизація). Це дозволяє значно скоротити розмірність простору. В результаті в якості ознак документа виступають всі значимі слова, що зустрічаються в документі.

Більшість сучасних алгоритмів машинного навчання орієнтовані на ознаковий опис об'єктів, тому всі документи зазвичай переводять в дійсний простір ознак. Для цього використовують ідею про те, що за приналежність документа до певного класу відповідають слова, а тексти з одного класу будуть використовувати багато схожих слів. Найбільш відомі способи, що дозволяють здійснити переклад тексту в простір ознак, засновані на статистичній інформації про слова. При їх використанні кожен об'єкт переводиться в вектор, довжина якого дорівнює кількості використовуваних слів у всіх текстах вибірки.

Наприклад, модель «мішка слів» (bag-of-words) дозволяє представити документ у вигляді багатовимірного вектора слів і їх ваг в документі. Іншими словами, кожен документ – це вектор в багатовимірному просторі, координати якого відповідають номерам слів, а

значення координат – значенням ваг. Основне припущення даного методу – порядок слів у документі не важливий, а колекцію документів можна розглядати як просту вибірку пар «документ-слово» (d, w), де d D, w W_d. У Bag of Words всі документи подаються у вигляді матриці T = (t) d, w, кожен рядок в якій відповідає окремому документу або тексту, а кожен стовпець – окремому слову. Елемент t_d, w відповідає кількості входжень слова w в документ d. Інша поширена модель індексації – Word2vec. Вона подає кожне слово у вигляді вектора, який містить інформацію про контекстні (супутні) слова. Часто інформацію в тексті несуть не тільки окремі слова, а й деяка послідовність слів. Наприклад, фразеологізми – стійкі поєднання слів значення яких не визначається значенням слів, що до них входять, взятих окремо. Наприклад, мовний зворот «Як риба у воді» означає відчувати себе впевнено, дуже добре в чому-небудь розбиратися. Сенс цього виразу буде передано невірно, якщо враховувати його слова окремо. Для того, щоб врахувати такі особливості мови пропонується при перекладі текстів у векторне подання враховувати крім слів, N-грами – послідовності із сусідніх символів.

Отриманий простір ознак буде сильно розрядженим і матиме високу розмірність за рахунок того, що різні слова зустрічаються у всій вибірці зазвичай багато. Через це для даного завдання найчастіше використовують лінійні методи машинного навчання.

Обчислювальна складність різних методів класифікації безпосередньо залежить від розмірності простору ознак. Тому для ефективної роботи класифікатора часто вдаються до скорочення числа ознак, що використовуються (термінів). За рахунок зменшення розмірності простору термінів можна знизити ефект перенавчання – явище, при якому класифікатор орієнтується на випадкові або помилкові характеристики навчальних даних, а не на важливі і значимі. Перенавчаний класифікатор добре працює на тих примірниках, на яких він навчався, і значно гірше на тестових даних. Щоб уникнути перенавчання, кількість навчальних

прикладів має бути пропорційно числу використовуваних термінів. У деяких випадках скорочення розмірності простору ознак в 10 разів (і навіть в 100) може призводити лише до незначного погіршення роботи класифікатора [15].

2.2 Огляд підходів та методів класифікації даних

Більшість підходів є ітераційними. Значний недолік багатьох підходів класифікації – теоретична необґрунтованість, відсутність доказів, що класифікація буде правильною, не кажучи вже про доведення оптимальності конкретного підходу.

Класифікація може здійснюватися повність вручну. У випадку, коли бібліотекар привласнює книгам певні тематичні рубрики. Проте, даний підхід не може застосовуватися, якщо необхідно розділити велику кількість документів на групи з високою швидкістю.

Інший підхід полягає в написанні правил, за якими відносять текст до однієї чи іншої групи. В правилі «якщо текст містить слова «похідна «і» рівняння», то віднести його до категорії математика». Спеціаліст, в якого є розуміння предметної області та навик написання регулярних виразів, може скласти ряд правил, які потім автоматично застосовуються до документів, що надходять, для їх класифікації. Цей підхід є кращим за попередній, оскільки процес категоризації автоматизується і, отже, кількість оброблюваних документів практично не обмежена. Більш того, побудова правил вручну може дати найкращу точність. Однак створення і підтримання правил в актуальному стані вимагає постійних зусиль фахівця.

Ще один підхід ґрунтується на машинному навчанні. Для вирішення завдання класифікації документів застосовуються методи машинного навчання з вчителем. Згідно з ними набір правил або критерій прийняття рішення текстового класифікатора, обчислюється автоматично з

навчальних даних, тобто здійснюється навчання класифікатора. Навчальні дані – це деяка кількість зразків документів з кожного класу. Приписування класу зразку-документу здійснюється вручну. Цей процес називають розміткою, що є більш простим завданням, аніж написання правил. Крім того, розмітка може бути проведена в звичайному режимі використання системи.

З найвідоміших методів класифікації виділяють наступні:

– метод найменших квадратів (метод регресійного аналізу);

– метод найближчих сусідів (метричний метод класифікації);

– метод опорних векторів (лінійний метод класифікації);

– метод Байеса (імовірнісний метод класифікації);

– метод дерев рішень (логічний метод класифікації).

Метод опорних векторів є лінійним методом класифікації. В даний час цей метод вважається одним з кращих методів класифікації. Будемо вважати множину документів, які необхідно класифікувати, множиною точок в просторі розмірністю D. Вибірку точок називають лінійно нероздільною, якщо точки, що належать різним класам, можна розділити за допомогою гіперплощини (в двовимірному випадку гіперплощиною є пряма лінія). Спосіб вирішення завдання в такому випадку – провести пряму так, щоб всі точки одного класу лежали по одну сторону від цієї прямої, а всі точки іншого класу були на протилежному боці. Тоді щоб класифікувати невідомі точки, досить буде подивитися, з якого боку прямої вони опиняться. Можна провести нескінченну множину гіперплощин (прямих), що задовольняють умові. Зрозуміло, що потрібно обрати пряму, максимально віддалену від наявних точок. У методі опорних векторів відстанню між прямою та множиною точок вважається відстань між прямою і найближчою до неї точкою з множини. Саме така відстань і максимізується в даному методі. Гіперплощина, що максимізує відстань до двох паралельних гіперплощин, називається поділяючою (на рисунку 2.2 позначена буквою L). Найближчі до паралельних гіперплощин точки

називаються опорними векторами (на рисунку 2.2 через них проходять пунктирні лінії). Алгоритм працює в припущенні, що чим більша різниця або відстань між цими паралельними гіперплощинами, тим менше буде середня помилка класифікатора, так як максимізація зазору між класами сприяє більш впевненій класифікації.

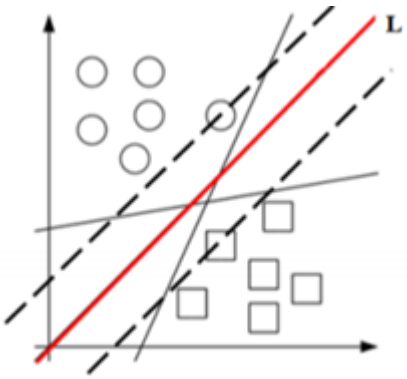


Рисунок 2.2 – Поділяюча гіперплощина в методі опорних векторів

На практиці структура даних буває невідома, і дуже рідко вдається побудувати поділяючу гіперплощину, а отже, неможливо гарантувати лінійну роздільність вибірки. Можуть існувати такі документи, які алгоритм відносить до одного класу, а в дійсності вони повинні належати до протилежного. Саме такі дані називаються викидами, адже вони призводять до похибки методу, тому було б краще їх ігнорувати. У цьому полягає сутність проблеми лінійної нероздільності. Вибірку називають лінійно нероздільною, якщо точки, що належать різним класам, не можна розділити за допомогою гіперплощини. Коли такої поділяючої гіперплощини не існує, здійснюють перехід від початкового простору ознак документів до нового, в якому навчальна вибірка виявиться лінійно нероздільною. Для цього кожний скалярний добуток замінюють на деяку функцію, що відповідає певним вимогам. Можна призначати певний штраф за кожний невірно класифікований документ. Цю функцію називають ядром. Заміна скалярного добутку функцією-ядром дозволяє

перейти до іншого простору ознак, де дані вже будуть роздільними. У разі лінійної нероздільності проблема пошуку оптимальної поділяючої гіперплощини зводиться до задачі, еквівалентній пошуку сідлової точки функції Лагранжа з умовами доповнюючої нежорсткості. Отримана система рівнянь вирішується методами квадратичного програмування. Це вже чисто обчислювальна задача. Цей варіант методу називають алгоритмом з м'яким зазором (soft-margin SVM), тоді як в лінійно роздільному випадку говорять про жорсткий зазор (hard-margin SVM).

Даний метод має як ряд недоліків так і переваг. Серед недоліків варто зазначити низьку швидкість навчання, складність інтерпретації параметрів методу, нестійкість по відношенню до викидів у вихідних даних. Проте, незважаючи на недоліки, метод є одним з найбільш якісних та потребує достатньо невеликого набору даних для навчання.

Метод дерев рішень відносять до логічних методів класифікації. Деревом рішень називають ациклічний граф, за яким здійснюється класифікація об'єктів (текстових документів), описаних набором ознак. Кожен вузол дерева містить умову розгалуження по одному з ознак. Кожен вузол має стільки розгалужень, скільки значень має обрана ознака. В процесі оцінки регулюються послідовні переходи від одного вузла до іншого відповідно до значень ознак об'єкта. Класифікація може вважатися завершеною, якщо було досягнуто одного з листів (кінцевих вузлів) дерева. Значення цього листа визначить клас, якому належить даний об'єкт. На практиці зазвичай використовують бінарні дерева рішень, в яких прийняття рішення переходу за ребрами здійснюється за допомогою звичайної перевірки наявності ознаки в документі. Якщо значення ознаки менше певного значення, то обирається одна гілка, якщо більше або дорівнює, – інша. На відміну від інших підходів, зазначених вище, даний підхід відноситься до символьних алгоритмів (рис. 2.3).

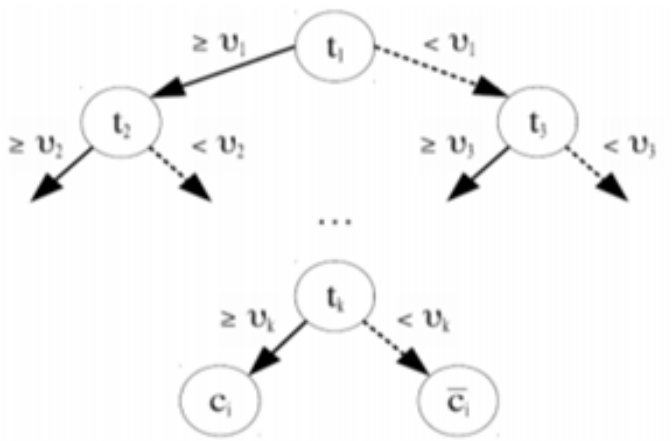


Рисунок 2.3 – Бінарне дерево рішень

Коли йде мова про вибір найбільш відповідної ознаки, як правило мають на увазі частотну ознаку, тобто будь-яку ознаку текста, що допускає можливість знаходження частоти його появи в тексті. Кращою для поділу є ознака, що дає максимальну на даному етапі інформацію про класи. Такою ознакою для тексту може бути, наприклад, ключове слово. З цієї точки зору будь-яку частотну ознаку можна вважати змінною. Тоді вибір між двома найбільш відповідними ознаками зводиться до оцінки ступеня зв'язності двох змінних. Тому для вибору підходящої ознаки на практиці застосовують різні критерії перевірки гіпотез, тобто критерії кількісної оцінки ступеня зв'язаності двох змінних, поставлених у взаємну відповідність: відповідає повній незалежності змінних, а 1 – їх максимальній залежності. Метод є нестійким по відношенню до викидів у вихідних даних і потребує великого обсягу даних для отримання точних результатів. Проте програмна реалізація методу відносно проста і результати роботи алгоритму легко піддаються інтерпретації [16].

Алгоритм найближчого сусіда (nearest neighbor) відносить об'єкт, що класифікується до того класу, якому належить найближчий навчальний об'єкт [17]. Цей метод відноситься до класу методів, робота яких ґрунтується на зберіганні даних в пам'яті для порівняння з новими елементами. При появі нового запису для прогнозування знаходяться

відхилення між цим записом і подібними наборами даних, і найбільш подібна (або ближній сусід) ідентифікується. Так як не завжди зручно зберігати всі дані, іноді зберігається тільки множина «типових» випадків. В такому випадку метод, що використовується, називають міркуванням за аналогією (CBR), міркуванням на основі аналогічних випадків, міркуванням за прецедентами. Прецедент – це опис ситуації в поєднанні з детальним зазначенням дій, що вживаються в даній ситуації. Підхід, заснований на прецедентах, умовно можна поділити на такі етапи:

– збір докладної інформації про поставлену задачу;

– зіставлення цієї інформації з деталями прецедентів, що зберігаються в базі, для виявлення аналогічних випадків;

– вибір прецеденту, найбільш близького до поточної проблеми, з бази прецедентів;

– адаптація обраного рішення до поточної проблеми, якщо це необхідно;

– перевірка коректності кожного знову отриманого рішення;

– занесення детальної інформації про новий прецедент в базу прецедентів.

Таким чином, висновок, заснований на прецедентах, становить такий метод аналізу даних, який робить висновки щодо даної ситуації за результатами пошуку аналогій, що зберігаються в базі прецедентів. Даний метод відноситься до категорії «навчання без вчителя», завдяки чому робочі характеристики кожної бази прецедентів з перебігом часу і накопиченням прикладів поліпшуються. Метод не створює будь-яких моделей або правил, узагальнюючих попередній досвід, у виборі рішення вони ґрунтуються на усьому масиві доступних історичних даних, тому неможливо сказати, на якій підставі будуються відповіді. Існує складність вибору міри «близькості» (метрики). Від цієї міри заходи головним чином залежить обсяг множини записів, які потрібно зберігати в пам'яті для досягнення задовільної класифікації або прогнозу. Крім того, існує висока

залежність результатів класифікації від обраної метрики. Метод не підходить для вирішення задач великої розмірності за кількістю класів і документів. Однак, використовуючи цей метод, не потрібно будувати класифікуючу функцію, а отже, є можливість оновлювати навчальну вибірку без перенавчання класифікатора. Алгоритм є стійким до аномальних викидів у вихідних даних. Програмна реалізація алгоритму відносно проста і результати роботи алгоритму легко піддаються інтерпретації.

Метод Байєса відноситься до імовірнісних методів класифікації. Наївна класифікація, або наївно-байесовский підхід (naive-bayes approach), є найпростішим варіантом методу, що використовує байєсовські мережі. Метод байєсівської класифікації є статистичним методом. Він дозволяє передбачити імовірність приналежності об'єкта до заданого класу. «Наївним» він називається тому, що виходить з припущення про взаємну незалежності ознак. Метод байєсівської класифікації заснований на теоремі Байеса:

(2.7)

де *H* – гіпотеза, яка полягає в тому, що об'єкт X належить до класу C;

P (H) – це імовірність a priori настання H;

P (H / X) – імовірність a posteriori виконання гіпотези H при даних X;

P (X / H), що спостерігаються – це імовірність a posteriori настання X за умови H.

Метод має відносно низьку якість класифікації. Не здатний враховувати залежність результату класифікації від поєднання ознак. Проте, байесовский метод дозволяє природним чином поєднувати закономірності, виведені з даних, і, наприклад, експертні знання, отримані в явному вигляді. Має високу швидкість роботи (зручний, коли накладаються жорсткі обмеження на час виконання класифікації, а

можливості скористатися більш точними методами немає), підтримку інкрементного навчання (класифікатор навчається на кожному окремо взятому зразку, немає необхідності пред'являти відразу всю навчальну вибірку). Програмна реалізація алгоритму відносно проста і результати роботи алгоритму легко піддаються інтерпретації (оскільки імовірності всіх ознак зберігаються, можна в будь-який момент подивитися, які ознаки документів є оптимальними для якісної класифікації) [18].

Метод найменших квадратів є методом регресійного аналізу, який використовується для оцінки невідомих параметрів за вибірковими даними. Основна суть методу – мінімізація суми квадратів відхилення між розрахунковими даними і емпіричним записом формули. Це один з найбільш поширених і розроблених в силу своєї простоти і універсальності метод. Нестійкість алгоритму по відношенню до викидів у вихідних даних та необхідність великого обсягу даних для отримання точних результатів є недоліками методу. Однак, програмна реалізація алгоритму відносно проста і результати роботи алгоритму легко піддаються інтерпретації [19].

Крім вищезазначених методів класифікації, використовують нейронні мережі. В якості вхідних даних обирається вектор параметрів єдиного об'єкта. Результатом роботи мережі буде код класу, до якого належить поданий на вході об'єкт.

2.3 Оцінка якості класифікації

При вирішенні задачі класифікації текстів типовою є ситуація, коли необхідно отримати деякі оцінки якості категоризації, які можна буде використовувати для порівняння різних методів і оптимізації параметрів методу. На дві частини розділяють колекцію категорізованих документів для оцінки якості: тестову, тобто перевірочну множину і навчальну, тобто тренувальну множину. Алгоритм навчання здійснюється на тренувальній множині. Навчений алгоритм застосовують до тестової множини і

розраховують на його основі метрики якості категоризації. Якість категоризації залежить від того, яким чином було здійснено розподіл множини категорізованих документів на тестову і навчальну множину. Якість побудованого класифікатора оцінюється за допомогою його помилки на тестовій підмножині навчальної множини документів. Під помилкою мається на увазі частка неправильно прийнятих рішень класифікатором. Отримані рішення класифікатора порівнюють з рішеннями експертів, які формують навчальну множину [20].

Основним критерієм при оцінці якості класифікації є комбінація точності і повноти. Найпростішим чисельним показником якості класифікації може бути відношення правильно класифікованих об'єктів до загальної їх кількості. Таку характеристику називають точністю (accuracy), яка задається формулою:

(2.8)

де P – кількість документів, за якими класифікатор прийняв правильне рішення;

N – розмір вибірки.

Зрозуміло, що ідеальний класифікатор має точність, рівну одиниці. Ця метрика має один істотний недолік: вона не враховує розподіл класів в навчальній вибірці. Тобто якщо для якихось класів у вибірці істотно більше даних, то для них класифікатор буде працювати відмінно, а для класів з меншою часткою інформації – гірше. Наприклад, точність класифікатора дорівнює 0,9 (що дуже добре), але в рамках якогось одного класу точність може бути менше 0,3. Звичайно, можна штучно збалансувати навчальну вибірку, вирівнявши співвідношення класів (наприклад, для чотирьох класів в навчальній вибірці повинно бути приблизно по 25% об'єктів кожного класу). Для проведення такої операції потрібно визначити клас з найменшим числом представників і взяти

стільки ж представників інших класів. Недоліком такого підходу є втручання в початковий набір даних, тому що інформація про частку об'єктів кожного класу сильно спотвориться. Тому точність хоч і використовується для оцінки якості, але повинна бути або підтверджена іншими характеристиками, або оцінена адекватно структурі навчальних даних. Точність (precision) і повнота (recall) є ключовими показниками якості класифікації (і не тільки класифікації). Найчастіше вони використовуються самі по собі, але іноді входять в якості основи для похідних метрик, таких як F-міра або R-Precision.

Точністю в межах класу називають відношення правильно класифікованих об'єктів до загальної кількості об'єктів, які класифікатор відніс до цього класу. Вона визначається за формулою:

(2.9)

де *TP* – істинно позитивні рішення;

FP – хибнопозитивні рішення.

Повнота визначається як частка знайдених класифікатором об'єктів деякого класу щодо всіх об'єктів даного класу в тестовому наборі за формулою [21]:

(2.10)

де *FN* – хибнонегативні рішення.

Для невеликої кількості класів існує наочний спосіб оцінки якості класифікації – матриця неточностей (confusion matrix). Звичайно, обмеження у вигляді невеликої кількості класів носить формальний характер, але користувачеві навряд чи буде зручно працювати з матрицею розмірності більше 100 елементів. Матрицею неточностей є матриця розміру N x N, де N – це число класів. За рядками матриці можна закріпити

істинні значення вихідної вибірки (завідомо правильна класифікація), а за стовпцями – результат роботи досліджуваного класифікатора (це також не є обмеженням, стовпці і рядки можна поміняти місцями, суть від цього не зміниться). Якщо класифікатор відніс об'єкт до класу m, а його істинний клас – n, то ми збільшуємо лічильник на перетині стовпця n і рядку m. Таким чином, легко зрозуміти, що якщо m і n збігаються, то ми потрапляємо на діагональний елемент, в цьому і є основний сенс матриці неточностей: ідеальний класифікатор має ненульову діагональ і інші елементи, рівні нулю. На рисунку 2.4 наведено приклад матриці неточності для шести класів.

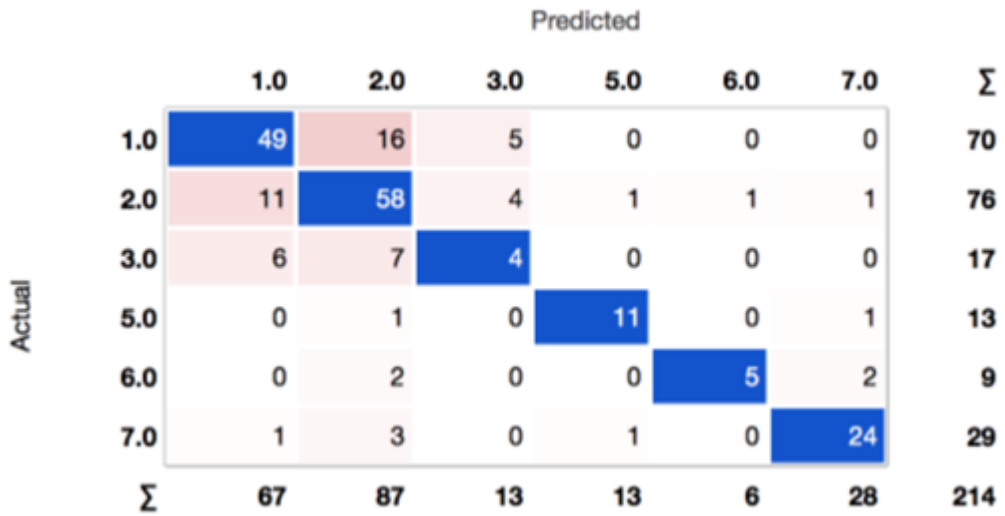


Рисунок 2.4 – Матриця неточностей для шести класів

Видно, що класифікатор добре впорався зі своїм завданням, однак у нього виникли проблеми з першими трьома класами (число невірно класифікованих об'єктів для них є найбільшим). Маючи цю матрицю можна розрахувати точність і повноту для всіх класів. Точність буде дорівнювати відношенню діагонального елемента матриці до суми елементів всього стовпчика класу. Повнота – відношення діагонального елемента матриці до суми елементів всього рядку класа. Узагальнену

точність і повноту класифікатора можна розрахувати як середнє арифметичне цих значень для всіх класів.

Чим ближче точність і повнота до одиниці, тим краще обраний і налаштований класифікатор. Для реальних даних і класифікаторів часто можна одночасно досягти і високої точності, і повноти. Для цього ввели похідну характеристику, яку назвали F-міра. F-міра – це гармонійне середнє між точністю і повнотою, представлене формулою[22]:

(2.11)

Середнє гармонійне має важливу властивість – воно прагне до нуля, якщо хоча б один з аргументів також прагне до нуля. Саме тому воно є кращим у порівнянні із середнім арифметичним (якщо алгоритм буде відносити всі об'єкти до позитивного класу, то він буде мати recall = 1 і precision «1, а їх середнє арифметичне буде більше 1/2, що неприпустимо). Вказану формулу розрахунку F-міри можна модифікувати, щоб точність і повнота мали різний внесок. Таким чином, можна розраховувати F міру з пріоритетом точності або з пріоритетом повноти. Формулу розрахунку F-заходи з використанням вагового коефіцієнта можна записати у вигляді:

(2.12)

де β приймає значення в діапазоні 0 <β <1, якщо ви хочете віддати пріоритет точності, а при β> 1 пріоритет віддається повноті. При β = 1 формула зводиться до попередньої, і ми отримуємо збалансовану F-міру (також її називають F1). Якщо β^2 = 0,25, F-міра вийде з пріоритетом точності, якщо β^2 = 2, то отримаємо F-міру з пріоритетом повноти. F-міра є дієвою інтегральною характеристикою якості класифікації, так як пов'язує точність і повноту.

Оцінка ефективності класифікаторів документів скоріше носить експериментальний характер, аніж аналітичний. Причиною тому є неформалізованності і суб'єктивність завдання текстової класифікації. Тому при експериментальній оцінці класифікаторів зазвичай визначають не складність алгоритму класифікатора, а його ефективність, тобто здатність правильно розподіляти документи за категоріями [23].

2.4 Навчання класифікатора побудованого на штучній нейронній мережі

Навчання нейронної мережі – це процес, в якому параметри нейронної мережі налаштовуються за допомогою моделювання середовища, в яку ця мережа вбудована. Тип навчання визначається способом підлаштування параметрів. Розрізняють алгоритми навчання з вчителем і без вчителя. Процес навчання з вчителем є висунення мережі вибірки навчальних прикладів. Останнім часом все більш популярними стають нечіткі класифікатори, тобто класифікатори, в процесі функціонування або навчання яких використовуються нечіткі множини. Сьогодні найбільш часто застосовуються класифікатори на основі логічного виводу за базою продукційних правил, антецеденти яких містять нечіткі терми «низький», «середній», «високий» і т.д. Кожне правило описує область факторного простору, всередині якої об'єкти належать одному класу. Межі цих областей нечіткі, тому один і той же об'єкт може одночасно належати декільком класам, але з різним ступенем.

Основні переваги нечітких класифікаторів обумовлені тим, що:

– логічний вивод за базою нечітких правил є прозорим;

− моделі класифікації компактні – для опису складних розділяючих поверхонь необхідно лише кілька лінгвістичних правил;

− формування бази лінгвістичних правил зазвичай не викликає труднощів у експерта;

– логічний вивод можна реалізувати як для числових, так і для категоріальних і нечітких значень вхідних ознак. При цьому в алгоритмі логічного виводу модифікується лише процедура фазифікації, а сама модель класифікації не змінюється.

Зазначені переваги дозволяють нечітким моделям прийняття рішень успішно конкурувати з класифікаторами на основі байесовских правил, методу найближчих сусідів, машини опорних векторів, нейронних мереж та інших методів індуктивної обробки даних. Для підвищення безпомилковості нечіткий класифікатор навчають на експериментальних даних. Існують два підходи до навчання нечіткого класифікатора. Перший підхід заснований на структурній ідентифікації залежності «входи-вихід» за допомогою нечітких правил. Він полягає у формуванні бази правил зі списку-кандидатів, виборі лінгвістичних квантифікаторов, наприклад, «дуже», «більш-менш», для термів антецедентів правил. Навчання в цьому випадку зводиться до вирішення дискретної задачі оптимізації. Другий підхід заснований на параметричній ідентифікації залежності «входи-вихід» за допомогою нечітких правил. При цьому під час навчання семантика правил не змінюється, а модифікуються функції приналежності нечітких термів і вагові коефіцієнти правил. Навчання в цьому випадку зводиться до вирішення задачі оптимізації з безперервними керованими змінними [24].

3 ОПИС АЛГОРИТМУ І АНАЛІЗ ОБРАНОЇ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ КЛАСИФІКАЦІЇ ТЕКСТІВ

3.1 Застосування мереж Кохонена

3.1.1 Самоорганізовані карти Кохонена

Нейронні мережі Кохонена або самоорганізовані карти Кохонена (Kohonen's Self-Organizing Maps) призначені для вирішення завдань автоматичної класифікації, коли навчальна послідовність образів відсутня. Відповідно відсутня і фіксація помилки, на мінімізації якої засновані алгоритми навчання, наприклад, алгоритм зворотнього поширення помилки. Мережа Кохонена – це двошарова нейронна мережа, яка містить вхідний шар (шар вхідних нейронів) і шар Кохонена (шар активних нейронів). Шар Кохонена може бути: одновимірним, двовимірним або тривимірним. У першому випадку активні нейрони розташовані в ланцюжок. У другому випадку вони утворюють двовимірну сітку (зазвичай у формі квадрата або прямокутника), а в третьому випадку вони утворюють тривимірну конструкцію. В силу відсутності навчальної послідовності образів, для кожного з яких відома від вчителя приналежність до того чи іншого класу, визначення ваг нейронів шару Кохонена засноване на використанні алгоритмів класичної класифікації (кластеризації або самонавчання).

На рисунку 3.1 наведено приклад топологічної карти мережі Кохонена, що містить вхідний шар і шар Кохонена. Нейрони вхідного шару слугують для введення значень ознак образів, що розпізнаються. Активні нейрони прошарку Кохонена призначені для формування областей (кластерів) різних класів образів. На цьому рисунку показані зв'язки всіх вхідних нейронів лише з одним нейроном шару Кохонена. Кожен нейрон шару Кохонена також з'єднаний з сусідніми нейронами [25].

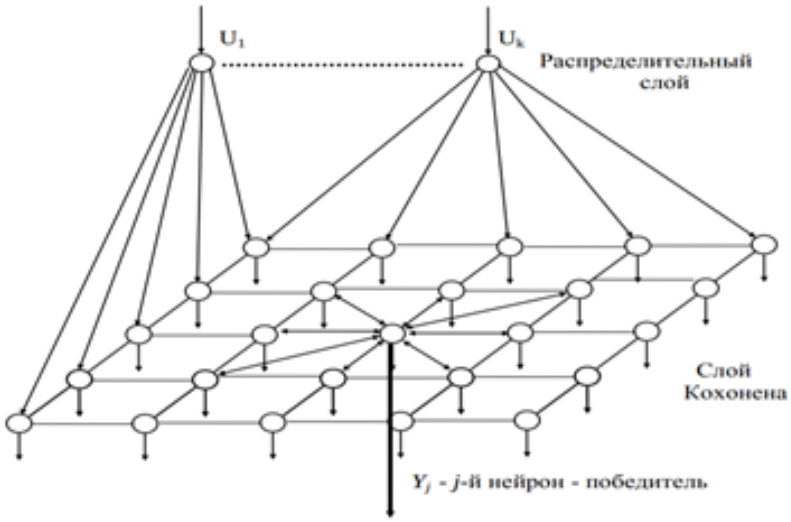


Рисунок 3.1 – Топологічна карта мережі Кохонена

На стадії навчання (точніше самонавчання) мережі вхідний вектор X_c попарно порівнюється з усіма векторами Wj всіх нейронів шару Кохонена. Вводиться деяка функція близькості (наприклад, у вигляді евклідової відстані). Активний нейрон з номером шару Кохонена, для якого значення функції близькості d (X, W_c) між вхідним вектором X, що характеризує певний образ, і вектором W_c максимально, оголошується «переможцем». При цьому образ, який характеризується вектором X, відноситься до класу, який є «нейроном-переможцем». В результаті здійснюється перетворення n-мірного вхідного простору R_n на m-мірну сітку (шар Кохонена). Слід підкреслити, що це відображення реалізується в результаті рекурентної (ітераційної) процедури самонавчання. Відмінна особливість цього відображення – формування кластерів або класів.

По завершенні процесу самонавчання на стадії реального використання мережі Кохонена невідомі вхідні образи відносяться до одного з виявлених кластерів (класів). На стадії самонавчання мережі Кохонена здійснюється корекція вагового вектора не тільки «нейрона-переможця», а й вагових векторів інших активних нейронів шару Кохонена, проте в значно меншій мірі – в залежності від віддалення від

«нейрона-переможця». При цьому форма і величина округа навколо «нейрона-переможця», вагові коефіцієнти нейронів якої також коригуються, в процесі навчання змінюються. Спочатку починають з дуже великою області - вона, зокрема, може включати всі нейрони прошарку Кохонена [26].

3.1.2 Мережа векторного квантування

Завдання векторного квантування з k кодовими векторами W_j для заданої сукупності вхідних векторів X ставиться як задача мінімізації перекручування при кодуванні, тобто при заміщенні кожного вектора з X відповідним кодовим вектором. У базовому варіанті мереж Кохонена використовується метод найменших квадратів і перекручування D обчислюється за формулою:

(3.1)

де K_j складається з тих точок x належить X, які ближче до W_j, ніж до інших W_i ($i \neq j$). Іншими словами, K_j складається з тих точок x належить X, які кодуються кодовим вектором W_j. Якщо сукупність X задана і зберігається в пам'яті, то стандартним вибором в навчанні відповідної мережі Кохонена є метод K-середніх. Це метод розщеплення:

– при даному виборі кодових векторів (вони ж вагові вектори мережі) W_j мінімізацією D знаходимо множини K_j, які складаються з тих точок x належить X, які ближче до W_j, аніж до інших W_i;

– при цьому розбитті X на множини K_j мінімізацією D знаходимо оптимальні позиції кодових векторів W_j, для оцінки за методом найменших квадратів це просто середні арифметичні:

(3.2)

де модуль K_j – кількість елементів в Kj.

Далі ітеруемо. Цей метод розщеплення сходиться за кінцеве число кроків і дає локальний мінімум перекручування. Якщо ж, наприклад, сукупність X заздалегідь не задана, або з певних причин не зберігається в пам'яті, то широко використовується онлайн метод. Вектори вхідних сигналів x обробляються по одному, для кожного з них знаходиться найближчий кодовий вектор («переможець», який «забирає усе») W_j (x). Після цього даний кодовий вектор перераховується по формулі:

(3.3)

де – це шаг навчання.

Інші кодові вектори на цьому кроці не змінюються [27].

Квантування і зниження розмірності можна розглядати як окремий випадок задачі регресії: простір відповідей збігається з простором ознак, і потрібно наблизити тотожне відображення за допомогою відображень виду c ◦ f, множина значень яких або кінцева (квантування), або маломірна (зниження розмірності) [28].

3.2 Обґрунтування обраного типу нейронної мережі

Нейромережеві класифікатори відрізняються один від одного за способом формування областей прийняття рішень.

Використання імовірнісних класифікаторів передбачає апріорне знання розподілів імовірностей для вхідних характеристик. Найчастіше використовується гаусовий розподіл або сума гаусових розподілів. Параметри розподілів, як правило, визначаються в процесі навчання з вчителем, при цьому всі дані для навчання повинні бути доступні одночасно. Ці класифікатори забезпечують оптимальну якість

класифікації, якщо розподіли, що використовуються є точною моделлю тестових даних і є доступним достатня для точного визначення параметрів кількість навчальних даних. Ці дві умови часто не виконуються для нестаціонарних систем і даних реального світу (результатів вимірювань).

Гіперплощині класифікатори формують складні області прийняття рішень, використовуючи вузли, які формують межі прийняття рішень у вигляді гіперплощин в просторі входів. Як правило, вузли обчислюють нелінійну функцію від зваженої суми вхідних значень. Найчастіше використовується сигмоподібна не лінійність, але також використовуються інші не лінійності, зокрема поліноми високого порядку. Ці класифікатори мають малу обчислювальну складність і вимагають мало пам'яті в процесі класифікації, але можуть вимагати багато часу для навчання. Сюди входять багатошарові персептрони, машини Больцмана, що класифікують бінарні дерева, мережі високого порядку, мережі, що формуються методом Group Method of Data Handling (GMDH) [29].

Класифікатори мережі прямого поширення складаються з декількох шарів нейронів: вхідного шару, вихідного і декількох «прихованих» шарів. Нейрони кожного шару не пов'язані між собою. Вихідний сигнал з кожного нейрона надходить на входи всіх нейронів наступного шару. Нейрони вхідного шару не здійснюють перетворення вхідних сигналів, їх функція полягає в розподілі цих сигналів між нейронами першого прихованого шару. Функціонування мережі прямого поширення надзвичайно просто. Вхідний сигнал, що подається на мережу, надходить в нейрони вхідного шару, проходить по черзі через всі шари і виділяється з виходом нейронів вихідного шару. У міру поширення сигналу по мережі він зазнає ряд перетворень, які залежать від його початкового значення, від перетворювальної функції і величин ваг зв'язків [30].

Класифікатори на основі LVQ-мережі, що заснована на векторному квантуванні (LVQ – Learning Vector Quantization), що навчається, становлять шар Кохонена, який навчається з вчителем. Для побудови LVQ-

мережі задається кількість кластерів (нейронів) n, кількість класів m (n m³) і приналежність кожного кластера певному класу. Розділити кластери за класами можна в тих же пропорціях, що і розподіл прикладів відповідних класів в навчальній вибірці. Відповідність між номерами кластерів і номерами класів може бути довільною. Для простоти інтерпретації результатів роботи мережі доцільно пронумерувати кластери послідовно. В процесі навчання LVQ-мережі ваги нейронів налаштовуються з урахуванням приналежності навчальних прикладів і кластерів одного класу. Навчена LVQ-мережу здійснює кластеризацію вхідних векторів з урахуванням класів (рис 3.2).

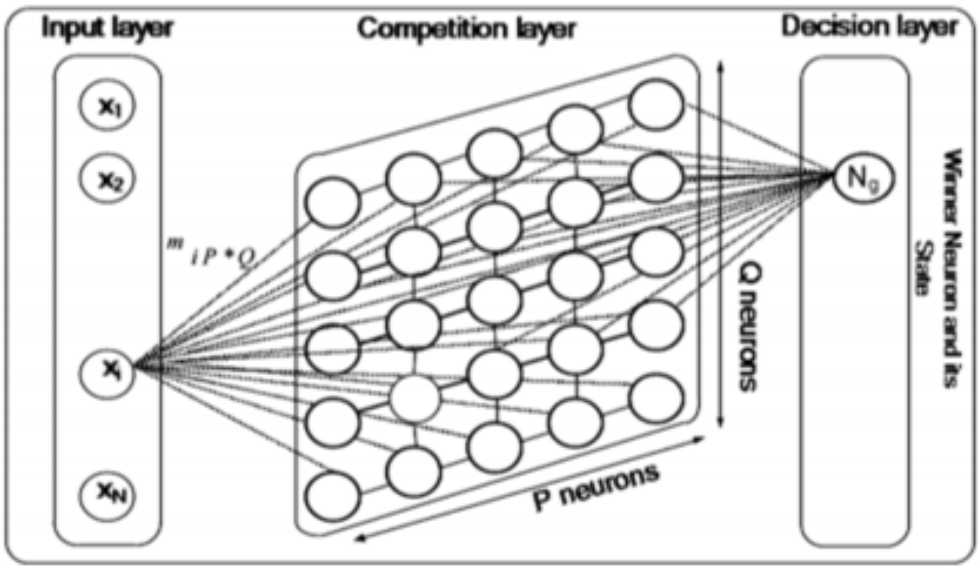


Рисунок 3.2 – Топологічна карта мережі Кохонена

Для класифікації текстів була обрана штучна нейронна мережа векторного квантування з вчителем. Дана мережа має досить просту одношарову архітектуру, налаштування семантичних ваг якої проводиться в режимі навчання з вчителем з елементами конкуренції за типом «переможець отримує все». Основними перевагами цієї ІНС в порівнянні з іншими нейросистемами є простота архітектури, незначна кількість

вхідних в неї нейронів, малий обсяг навчальної вибірки і можливість online навчання. До теперішнього часу відома велика кількість варіантів LVQ-нейромереж, що відрізняються вибором параметра кроку навчання, використовуваною метрикою, необхідним обсягом навчальної вибірки. Ці системи підтвердили свою ефективність у багатьох додатках, пов'язаних з чіткою класифікацією і розпізнаванням образів [31].

LVQ на етапах навчання та прийняття рішення обчислює відстань між прототипом вектора M_i кожного нейрона та вхідним вектором. На фазі прийняття рішення нейроном-переможцем є нейрон з найменшою відстаню. У фазі навчання, тільки ваги нейрона-переможця будуть змінені відповідно до правила теорії Хебба [32].

3.3 Особливості класифікації текстів новинної тематики

В епоху інформації новини є доступними через Інтернет-джерела, що спричиняє необхідність класифікації таких даних, як мають значний вплив на різні сфери нашого життя.

Новинні текстові дані іноді містять небачену раніше інформацію, тому мають потребу у динамічній класифікації та дослідженні. Оскільки класифікація повинна проводитися за допомогою розріджених даних для навчання, це обумовлює проблему для стандартних методів класифікації. Тому використовуються більш високоефективні алгоритми класифікації. Велика кількість новин пов'язана з різними категоріями, такими як спорт, технології, розваги, музика та політика. Кожен користувач може побачити такі типи новин в Інтернеті. Якщо користувача цікавлять новини, пов’язані з конкретною тематикою, він переходить до розділу цих новин, обираючи потрібну. Це дуже трудомісткий процес. Існує можливість відображати новини для користувача за його вибором. Оскільки кількість веб-сайтів значно зросла, користувачеві стало дуже важко отримувати новини за

власним інтересом. В результаті чого, актуальною стала фільтрація новинних текстів згідно за категоріями з метою швидкого доступа до них.

Кожна отримана новина складається з таких компонентів як заголовок новини, опис новини, посилання та метадані (автор або дата публікації). Отримані дані можуть бути неповними, шумними і непослідовними. Такі дані становлять певну проблему для якісного отримання результатів класифікації.

3.4 Алгоритм класифікації новинних текстів

Кожен з документів повинен бути представлений у вигляді вектора термінів, зрозумілого нейронній мережі. Щоб виділити безліч термінів документа, необхідно провести його деяку «чистку» (прибрати цифрові символи, знаки пунктуації, «стоп-слова» – слова загальної лексики, прийменники, сполучники, частки). Таким чином, відкидаються слова, які не повинні впливати на результати пошуку, і залишаються терміни, що безпосередньо впливають на віднесення документа в будь-яку категорію. Кожен термін – це одна з ознак документа, а сукупність цих ознак є вектором всього документа (рис 3.3).

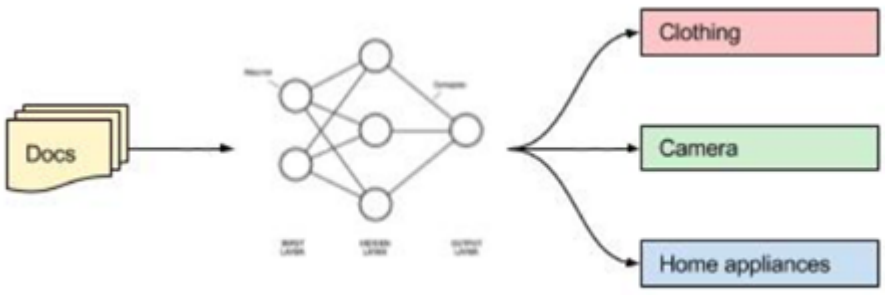


Рисунок 3.3 – Загальна архітектура класифікації

Попередня обробка текстів починається з токенізації – розбиття послідовності тексту на символи, фрази, слова та всі інші значущі

елементи на лексеми. На початку текстові дані є лише блоком символів, що перетворюються у набір слів (рис. 3.4).

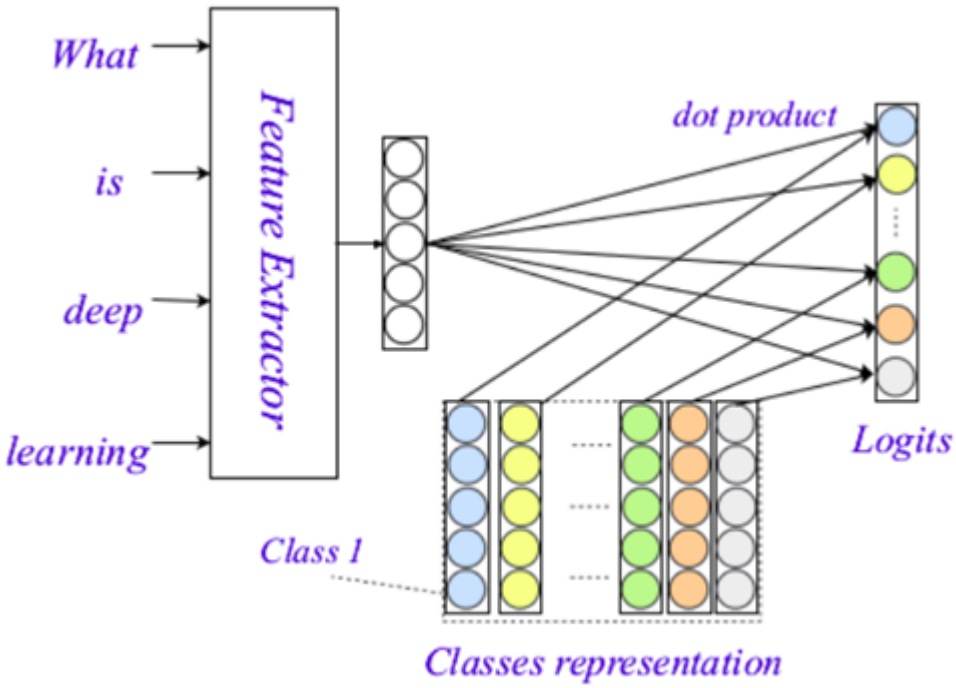


Рисунок 3.4 – Текстова класифікація

Основне використання токенізації – це виявлення значимих ключових слів. Деякі дуже поширені слова, які не несуть ніякого смислового значення – стоп-слова видаляються з лексики повністю. Іноді також використовується стемінг – приведення слова до основної форми. Слова, що мають різну відміну несуть один і той же зміст.

Для покращення точності результатів класифікації можна розширювати набір слів словами-синонімами за допомогою бази даних WordNet.

Після попередньої обробки тексту, необхідно перевести текстові дані у вектор, довжина якого дорівнює кількості використовуваних слів у всіх текстах вибірки. Bag of words це найбільш широко використовуємий спосіб векторного подання тексту. Колекція документів розглядається як проста вибірка пар «документ-слово» (d, w), де d D, w W d. У Bag of

Words всі документи представлені у вигляді матриці T = (t) d, w, кожен рядок в якій відповідає відповідному документу або тексту, а кожен стовпець – окремому слову. Елемент t d, w відповідає кількості входжень слова w в документ d. Для оцінки важливості слова в документі використовується міра tf-idf:

. (3.4)

TF – частота слова, оцінює важливість слова w_i в межах окремого документа:

, (3.5)

де n_i – число входжень слова і в документ;

$\sum k\ n_k$ – загальна кількість слів у даному документі.

IDF – зворотна частота документа. Облік IDF зменшує вагу широко вживаних слів:

(3.6)

де | D | – кількість документів в корпусі.

– кількість документів, в яких зустрічається слово w_i.

Отже, алгоритм класифікації новинних текстів складається з наступних етапів:

1) аналіз існуючих джерел текстових даних, вибір датасету та побудова власної інформаційної бази;

2) попередня обробка та переведення новинних текстів у векторний вигляд для нейронної мережі за допомогою статистичних мір, що успішно використовуються в області NLP;

3) обґрунтування і вибір архітектури нейронної мережі;

4) подання на вхід вектору ознак документів нечіткій нейронній мережі;

5) отримання результатів класифікації;

6) перевірка отриманих результатів за допомогою мір оцінки якості класифікації.

4 РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

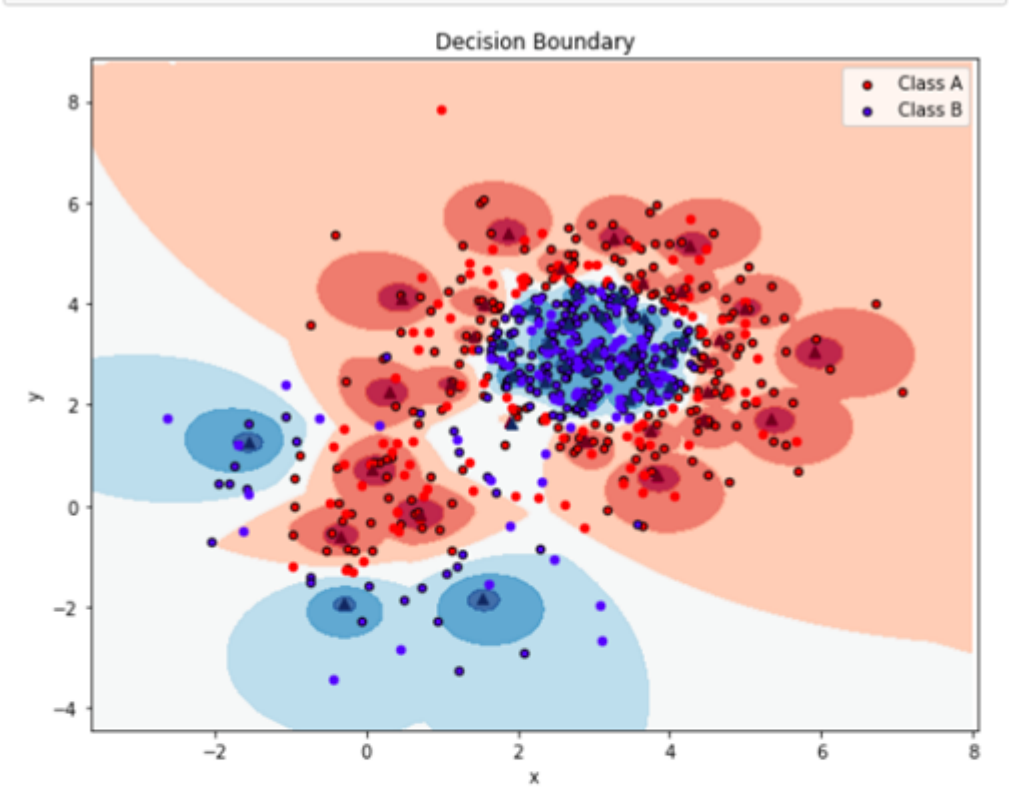
4.1 Опис контрольного прикладу класифікації новинних текстів

В якості інформаційної бази було обрано корпус англомовних новинних текстів AG News, який містить новини з Інтернету, що відносяться до чотирьох найбільших класів, таких як world, sport, business та sci / tech [https://datasets.quantumstat.com/]. Даний датасет містить 30 000 навчальних прикладів та 1900 прикладів тестування для кожного класу (рис 4.1).

Class Index	Title	Description
Consists of class ids 1-4 where 1-World, 2-Sports, 3-Business, 4-Sci/Tech	Contains title of the news articles	Contains description of the news articles
1 — 4	7568 unique values	7594 unique values
3	Fears for T N pension after talks	Unions representing workers at Turner Newall say they are 'disappointed' after talks with stricken...
4	The Race is On: Second Private Team Sets Launch Date for Human Spaceflight (SPACE.com)	SPACE.com - TORONTO, Canada -- A second\team of rocketeers competing for the #36;10 million Ansari ...
4	Ky. Company Wins Grant to Study Peptides (AP)	AP - A company founded by a chemistry researcher at the University of Louisville won a grant to deve...

Рисунок 4.1 – Інформаційна база новинних текстів

Класифікатор написаний на мові програмування Python, оскільки дана мова містить значну кількість бібліотек з алгоритмами нейронних мереж. На рисунку 4.2 відображено результат класифікації за допомогою системи LVQ з інформацією про точність класифікації, де кожен клас (червоний і синій) представлений одним прототипом.




```
Train data
        Accuracy: 0.945
        Brier: 0.080
        Precision (Efficiency): 0.951
        Recall (Completeness): 0.946
        F1: 0.949

Test data
        Accuracy: 0.900
        Brier: 0.104
        Precision (Efficiency): 0.863
        Recall (Completeness): 0.908
        F1: 0.885
```

Рисунок 4.2 – Класифікація

На першому етапі необхідно вилучити дані необхідні для класифікації, для цього загружається інформаційна база текстів (рис 4.3).

```
import numpy as np
import pandas as pd


class NewsDataset:
    @classmethod
    def load_dataset_and_vectorize(cls, news_dataset):
        news_dataset = pd.read_csv(news_dataset)
        return cls(news_dataset)
```

Рисунок 4.3 – Загрузка даних з датасету

Після попередньої обробки даних за допомого стеммінга чи лематизації, видалення стоп-слів, переведення отриманих наборів слів у векторний простір, вектор ознак тексту подається на вхід нейронної мережі, що зображено на рисунку 4.4.

```
network = nl.net.newlvq(nl.tool.minmax(data), 10)

error = net.train(data, labels, epochs=100, goal=1)
```

Рисунок 4.4 – Ініціалізація процесу тренування мережі

На наступному рисунку 4.5 відображається процес класифікації.

Дані інформаційної бази розділені на тестову вибірку та вибірку для навчання.

На тестовій виборці оцінюється якість побудованого класифікатора. Якщо навчальна і тестова вибірки незалежні, то оцінка, зроблена за тестовою вибіркою, є незміщеною (рис. 4.6).

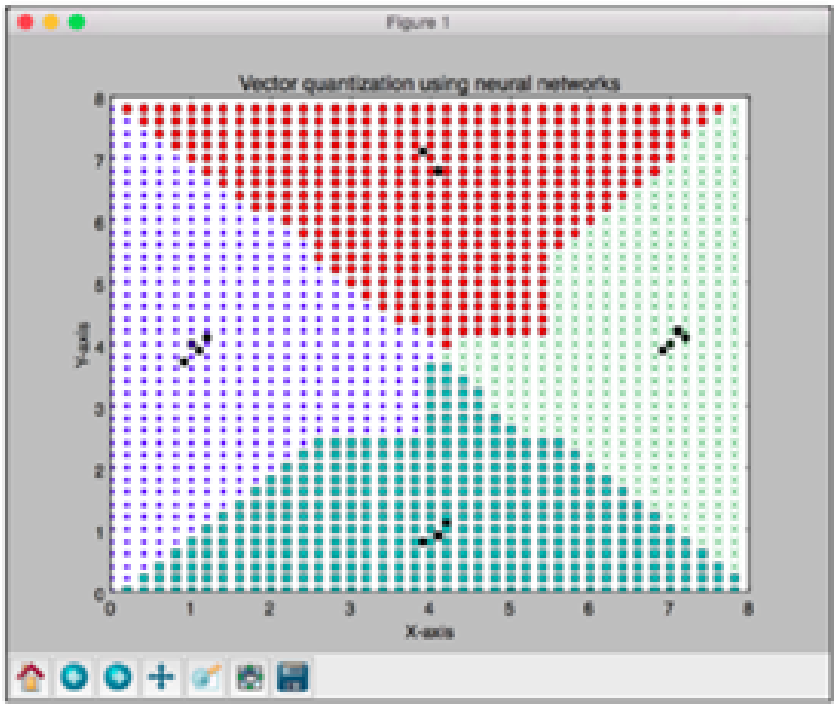


Рисунок 4.5 – Процес класифікації

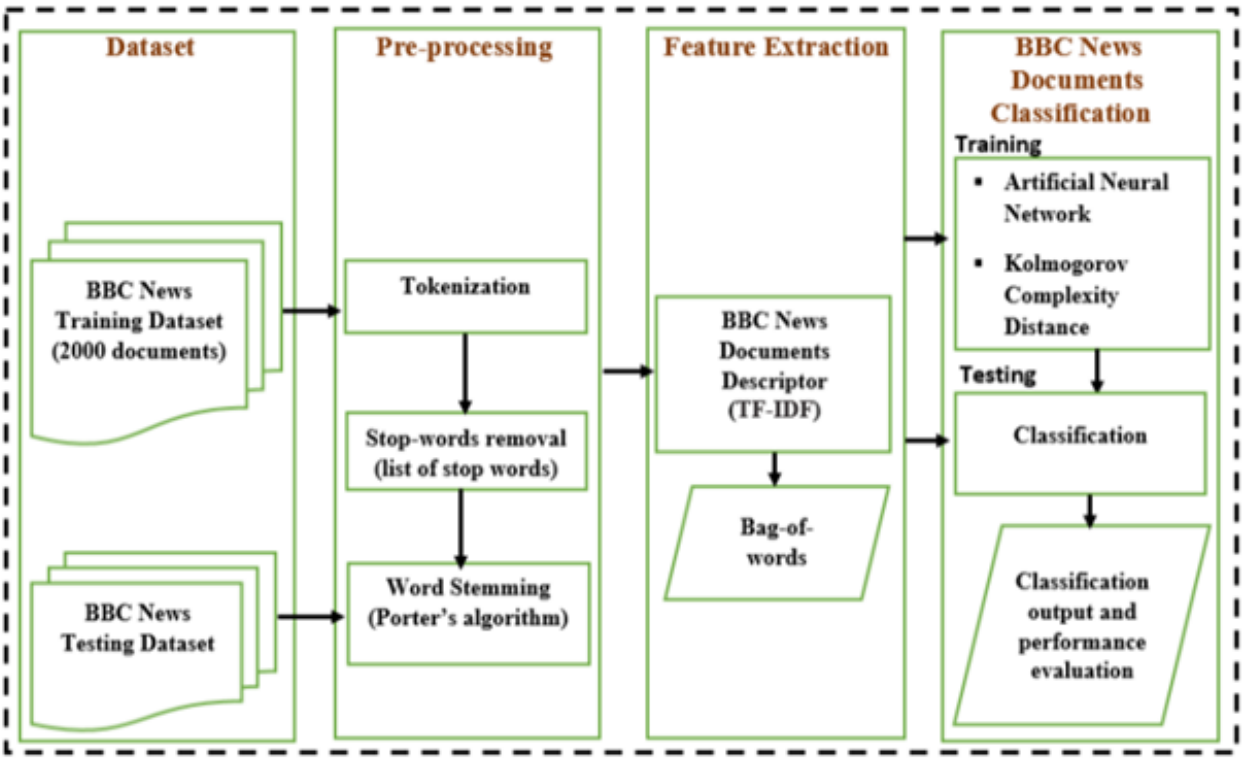


Рисунок 4.6 – Повний цикл класифікації даних

Оцінку якості, зроблену за тестовою вибіркою, можна застосовувати для вибору найкращої моделі. Однак тоді вона знову опиниться оптимістично зміщеною. Для отримання незміщеної оцінки обраної моделі доводиться виділяти третю вибірку – валідаційну, за якою здійснюється вибір найкращої моделі з множини моделей, побудованих за навчальною вибіркою.

4.2 Аналіз існуючих систем класифікації

На сьогодні в откритому доступі є низка систем, що здійснюють класифікацію текстів різних тематик в реальному часі. Звісно, що найчастіше це API, які хоч і мають демо версію, проте є платними. Серед класифікаційних систем новинної тематики було розглянуто online класифікатор Dandelion API (рис. 4.7) [33].

Дана система класифікує лише невеличкі частини текстової інформації і має досить точну класифікацію. Речення новинних текстів можуть містити різні ключові слова, що можуть відноситися до деяких категорій відразу і в такому випадку, система відображає відношення певного речення до декількох категорій одночасно у вигляді зафарбованих кольором областей на павутинні. Проте, незважаючи на цікавість даної системи, вона працює лише з певним невеликим параграфом новостної статті, великі в об’ємі текстові статті вона нажаль не оброблює. А саме така обробка має сенс в епоху великих масивів даних.

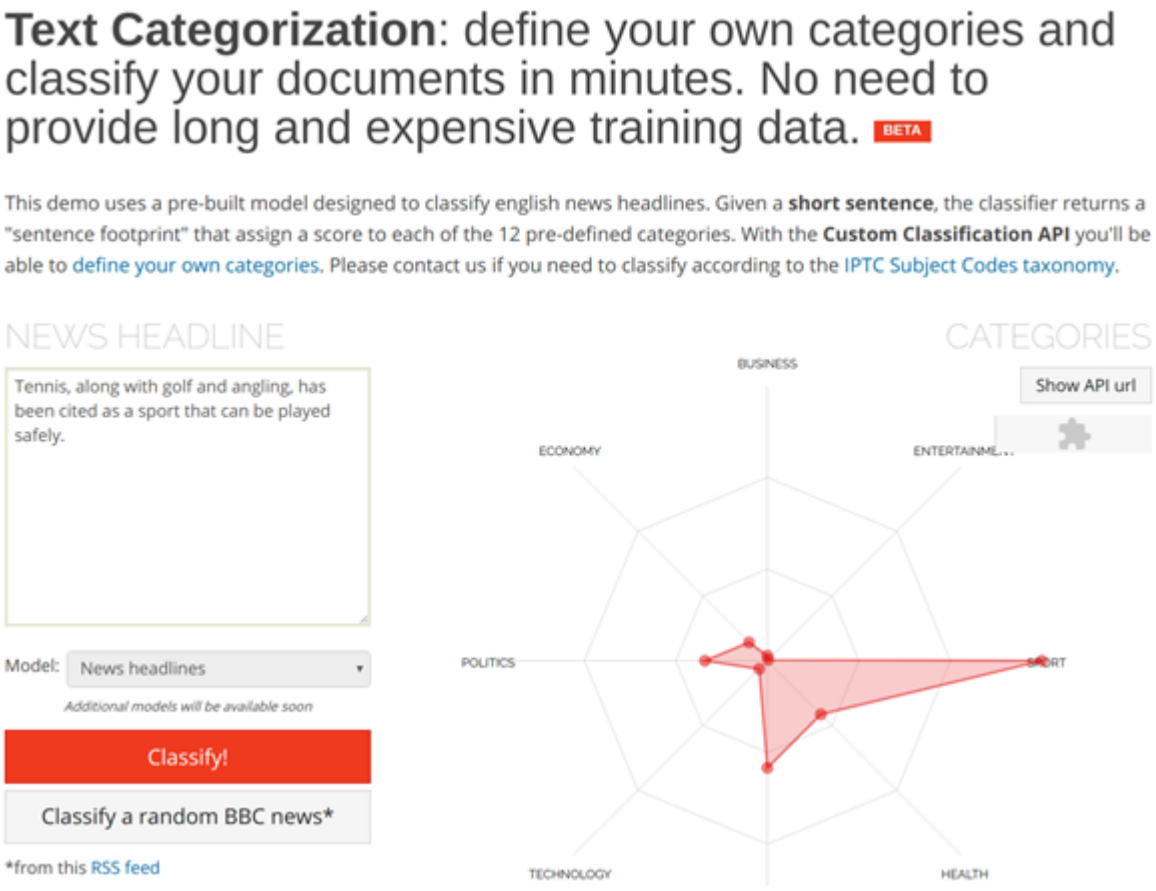


Рисунок 4.7 – Система класифікації новинних даних Dandelion API

Система ParallelDots також дає можливість класифікувати новинні тексти. В результаті класифікації система видає відсоток належності введеного речення до категорії (рис 4.8). Наприклад, для речення «Tennis, along with golf and angling, has been cited as a sport that can be played safely, while keeping two metres apart from anyone else.» зі статті, що належить до категорії Спорт, система видає 73, 9 % приналежності категорії Спорт та значний відсоток категорії Розваги. І дійсно, певні слова, такі як «angling» , «played» можно віднести до категорії Розваг [34].

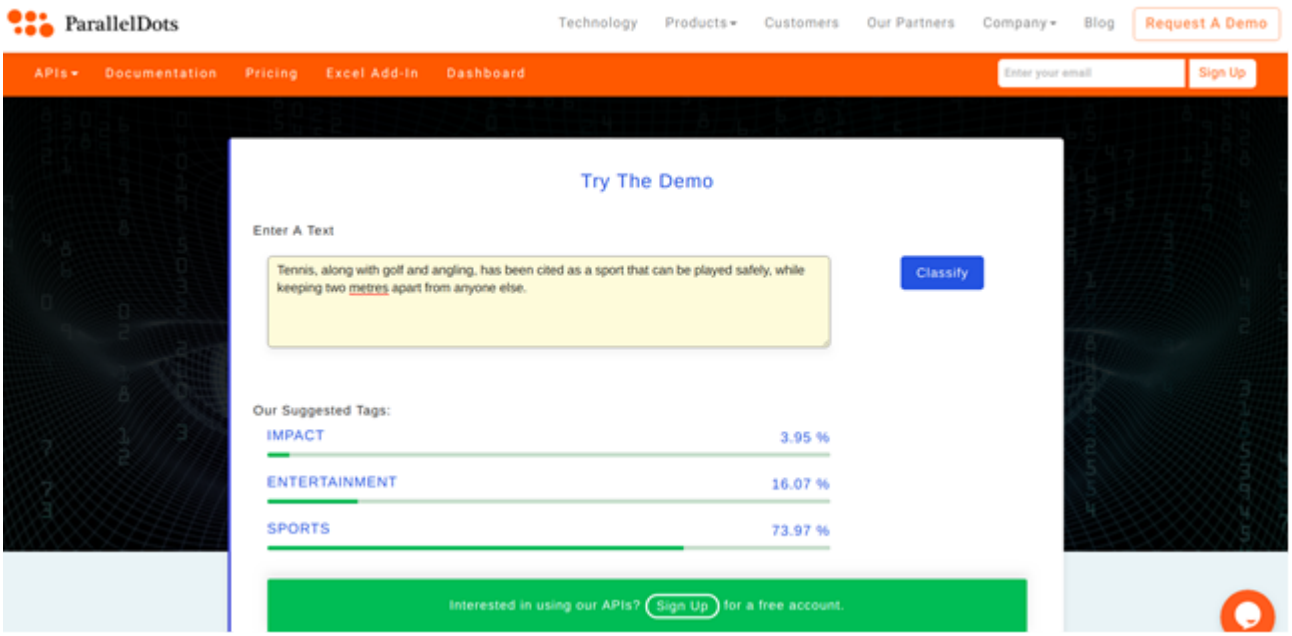


Рисунок 4.8 – Система класифікації текстових даних ParallelDots

Однак, без використання мір оцінки якості, дуже важко визначити точність класифікації, адже лише одне просте речення не дає повноцінного розуміння якості даної системи.

Розглянуті системи орієнтовані на невелику кількість даних і якщо за допомогою даних систем намагатися аналізувати категорію статті цілком, дані системи не виключають значних похибок категоризації.

4.3 Обґрунтування вибору мови програмування

В якості засобу реалізації програми була обрана об'єктно-орієнтована високорівнева мова програмування Python. Вбудовані високорівневі структури даних в поєднанні з динамічною типізацією і зв'язуванням роблять мову відповідною для швидкої розробки додатків.

Програмний код на Python перевершує інші високорівневі мови програмування за рахунок зручної читабельності та простішого виконання його використання та обслуговування. Python підтримує найсучасніші механізми багаторазового використання програмного коду, обсяг якого на

даній мові зазвичай становить третину або навіть п'яту частину еквівалентного програмного коду на мові C ++ або Java. Це істотно зменшує кількість часу на налагодження і потребує меншого обсягу трудовитрат на супровід. Крім того, програми на мові Python запускаються відразу ж, оминаючи тривалі етапи компіляції і зв'язування, необхідні в деяких інших мовах програмування.

У складі Python поставляється велика кількість зібраних і переносних функціональних можливостей, відомих як стандартна бібліотека. Ця бібліотека надає велику кількість можливостей, затребуваних в прикладних програмах, починаючи від пошуку тексту за шаблоном і закінчуючи мережевими функціями. Можливим є розширення бібліотеки за рахунок власних або сторонніх бібліотек. З числа сторонніх розробок можна відзначити інструменти створення веб-сайтів, програмування математичних обчислень, доступ до послідовного порту, розробку ігрових програм і багато іншого. Наприклад, розширення NumPy позиціонується як вільний і більш потужний еквівалент системи програмування математичних обчислень Mathlab. NumPy – за рахунок інтеграції з математичними бібліотеками, написаними на мовах програмування, що компілюються – перетворює Python в складний, але зручний інструмент програмування математичних обчислень, який найчастіше може замінити існуючий програмний код, написаний на традиційних мовах, що компілюються, таких як FORTRAN і C ++. Додаткові інструменти математичних обчислень для Python підтримують можливість створення анімаційних ефектів і тривимірних об'єктів. Серед незліченої кількості бібліотек особливу увагу акцентують на бібліотеці NLTK як на пакеті бібліотек і програм для символьної і статистичної обробки природної мови. Вона дозволяє здійснити первинну обробку текстової інформації через токенізацію, лематизацію, стемінг; містить методи для обчислень базової статистики тексту та розпізнавання іменованих сутностей.

В будовані в Python інтерфейси доступу до служб операційних систем роблять його ідеальним інструментом для створення переносних програм і утиліт системного адміністрування (іноді вони називаються інструментами командної оболонки). Програми на мові Python можуть відшукувати файли і каталоги, запускати інші програми, здійснювати паралельні обчислення з використанням декількох процесів і потоків. Стандартна бібліотека Python забезпечує можливість зв'язування відповідно до вимог стандартів POSIX і підтримує всі типові інструменти операційних систем: змінні оточення, файли, сокети, канали, процеси, багатопоточну модель виконання, пошук за шаблоном з використанням регулярних виразів, аргументи командного рядку, стандартні інтерфейси доступу до потоків даних та запуск команд оболонки. Крім того, системні інтерфейси в даній мові створені переносними, наприклад, сценарій копіювання дерева каталогів не вимагає внесення змін, в якій би операційній системі він не використовувався.

Python дозволяє писати дуже компактні і легкі для читання програми. Програми, написані на мові Python, зазвичай досить малооб'ємні, оскільки типи даних високого рівня в ньому дозволяють виразити складні операції однією інструкцією; групування інструкцій виконується за допомогою відступів замість фігурних дужок і немає необхідності в оголошенні змінних.

Для вирішення поставленого завдання необхідно було використовувати функціональну, ефективну і зручну платформу для розробки, що дозволяє застосовувати принципи об'єктно-орієнтованого програмування. В якості такої платформи було обрано інтегроване середовище розробки для мови програмування Python – PyCharm. PyCharm розробляється компанією JetBrains, відомою як IntelliJ IDEA та працює під операційними системами Windows, Mac OS X і Linux. Середовище має потужний і функціональний редактор коду з підсвічуванням синтаксису і помилок, авто-форматуванням і авто-відступами для підтримуваних мов;

потужну навігацію за проектами і вихідним кодом: відображення файлової структури проекту, швидкий перехід між файлами, класами та методами; вбудований відладчик для Python та інструменти для юніт-тестування; підтримку систем контролю версій: загальний користувальницький інтерфейс для Mercurial, Git, Subversion, Perforce і CVS з підтримкою списків змін і злиття. PyCharm дозволяє швидко здійснювати рефакторинг коду, а також використовувати зручний графічний відладчик.

ВИСНОВКИ

У ході виконання дипломної роботи були проаналізовані нечіткі нейронні мережі, їх особливості та застосування у тренуванні класифікатора, що визначає категорії для текстової інформації новин.

Для вирішення поставленого завдання проведено огляд підходів та методів класифікації даних, засобів оцінки якості класифікації, розглянуті особливості мережі векторного квантування. Було зроблено висновок, про те, що нейронні мережі, що навчаються з вчителем, є універсальними аппроксиматорами неперервних відображень «вхід вихід». Це дозволяє використовувати мережі для визначення відношення правдоподібності в задачах детектування і класифікації.

Завдяки нейронним мережам, сьогодні можна отримати якісні семантичні уявлення для слів, фраз і пропозицій, причому навіть без навчальної вибірки. Все менше зусиль зараз потрібно для створення власних семантичних словників і баз знань, тому розробляти системи автоматичної обробки текстів стало простіше. Однак ми все ще дуже далекі від адекватного вирішення завдання розуміння взаємопов'язаних подій, представлених у вигляді послідовності пропозицій або образів, а також діалогів. Всі відомі сьогодні методи успішно працюють або при вирішенні завдань «поверхневого» розуміння мови, або при істотному обмеженні предметної області.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1) Как работает нейронная сеть. URL: https://neurohive.io/ru/osnovy-data-science/osnovy-nejronnyh-setej-algoritmy-obuchenie-funkcii-aktivacii-i-poteri/ (дата звернення: 23.03.2020).

2) Рутковская Д., Пилинский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. Москва, 2013. 384 с.

3) Асадуллаев Р. Г. Нечеткая логика и нейронные сети: учебное пособие. Белгород, 2017. 309 с.

4) Вакуленко С. А., Жихарева А. А. Практический курс по нейронным сетям. СПб: Университет ИТМО, 2018. 71 с.

5) Каширина И. Л. Нейросетевые технологии: учебно-методическое пособие для вузов. Воронеж, 2008. 72 с.

6) Хайкин С. Нейронные сети. Москва: Вильямс, 2006. 1104 с.

7) Обработка естественного языка. URL: https://ru.wikipedia.org/wiki/Обработка_естественного_языка (дата звернення: 23.03.2020).

8) Близнюк Б. О., Васильева Л. В., Стрельников И. Д., Ткачук Д. С. Современные методы обработки естественного языка. Харьков, 2017. 13 с.

9) Головко В. А., Краснопрошин В. В. Нейросетевые технологии обработки данных: учебное пособие. Минск: БГУ, 2017. 263 с.

10) Ахметов Б. С., Горбаченко В. И., Кузнецова О. Ю. Нечеткие системы и сети: учебное пособие. Алматы: ТОО «Издательство LEM», 2014. 104 с.

11) Гольдберг Й. Нейросетевые методы в обработке естественного языка / пер. с англ. А. А. Слинкина. Москва: ДМК Пресс, 2019. 282 с.

12) Машинное обучение для понимания естественного языка. URL: https://www.osp.ru/os/2016/01/13048649/ (дата звернення: 23.03.2020).

13) Воронцов К. В. Курс лекций по машинному обучению. URL: https://www.MachineLearning.ru/ (дата звернення: 23.03.2020).

14) Анализ данных и процессов: учебное пособие / А. А. Барсегян и др. 3-е изд., перераб. и доп. СПб.: БХВ-Петербург, 2009. 166 с.

15) Батура Т. В. Методы автоматической классификации текстов. Новосибирск, 2016. 5 с.

16) Батура Т. В. Математическая лингвистика и автоматическая обработка текстов : учебное пособие. / Новосиб. гос. ун-т. Новосибирск: РИЦ НГУ, 2016. 166 с.

17) Воронцов К. В. Математические методы обучения по прецедентам (теория обучения машин). URL: https://www.MachineLearning.ru/ (дата звернення: 23.03.2020).

18) Чернышова Г. Ю. Интеллектуальный анализ данных: учебное пособие для студентов специальности «Прикладная информатика (в экономике)» / Саратовский государственный социально-экономический университет. Саратов, 2012. 92 с.

19) Применение метода наименьших квадратов в моделировании спроса и предложения на рынке образовательных услуг. URL: https://cyberleninka.ru/article/n/primenenie-metoda-naimenshih-kvadratov-v-modelirovanii-sprosa-i-predlozheniya-na-rynke-obrazovatelnyh-uslug/viewer (дата звернення: 23.03.2020).

20) Герасименко Е. М. Интеллектуальный анализ данных. Алгоритмы Data Mining : учебное пособие. Таганрог: Издательство Южного федерального университета, 2017. 84 с.

21) Чернышева Г. Ю. Интеллектуальный анализ данных: учебное пособие для студентов специальности «Прикладная информатика (в экономике)». Саратов : Саратовский государственный социально-экономический университет, 2012. 92 с.

22) Demšar J., Blaž Z. Orange: Data Mining Fruitful and Fun – A Historical Perspective. *Informatica*. 2013. Vol. 37. P. 55–60.

23) Анализ данных и процессов / А. Барсегян и др. 3-е изд. Санкт-Петербург: БХВ-Петербург, 2009. 512 с.

24) Штовба С. Д., Панкевич О.Д. Анализ критериев обучения нечеткого классификатора. URL: http://shtovba.vk.vntu.edu.ua/file/7bded12f79f54a64342fa156a4feba7d.pdf (дата звернення: 23.03.2020).

25) Круг П. Г. Нейронные сети и нейрокомпьютеры: учебное пособие по курсу «Микропроцессоры». Москва: Издательство МЭИ, 2002. 176 с.

26) Андреева К. А. Применение нейронной сети Кохонена для классификации web-страниц информационно-поисковой системой сайтов. URL: https://cyberleninka.ru/article/n/primenenie-neyronnoy-seti-kohonena-dlya-klassifikatsii-web-stranits-informatsionno-poiskovoy-sistemoy-saytov/viewer (дата звернення: 23.03.2020).

27) Шарипбай А. А. Нейронные сети: учебное пособие. Алматы: Эверс, 2017. 278 с.

28) Мерков А. Б. О статистическом обучении. URL: https://www.recognition.mccme.ru/pub/RecognitionLab.html/slt.pdf/ (дата звернення: 23.03.2020).

29) Овчинников П. Е. Применение искусственных нейронных сетей для обработки сигналов: учебно-методическое пособие. Нижний Новгород: Нижегородский госуниверситет, 2012. 32 с.

30) Бахтин А. В., Ремизова И. В. Элементы искусственного интеллекта в системах управления : учебное пособие. СПб., 2014. 54 с.

31) Бодянский Е.В., Рябова Н.В., Золотухин О.В. Обработка текстовых документов с помощью адаптивного нечеткого обучаемого векторного квантования. URL: http://openarchive.nure.ua/bitstream/document/3239/1/109-115.pdf (дата звернення: 23.03.2020).

32) Implementation of a novel LVQ neural network architecture on FPGA. URL: https://www.researchgate.net/publication/255738883_Implementation_of_a_novel_LVQ_neural_network_architecture_on_FPGA (last accessed: 23.03.2020).

33) Dandelion API. URL: https://dandelion.eu (last accessed: 23.03.2020).

34) ParallelDots API. URL: https://paralleldots.com (last accessed: 23.03.2020).

35) Christopher M. Bishop. Neural Networks for Pattern Recognition. Oxford University Press, 1995.

36) Breiman L., Friedman J., Stone C. J., Olshen R. A. Classification and Regression Trees. *Chapman & Hall/CRC*. January 1984.

37) Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика. Москва: Мир, 1992. 240 с.

38) Комарцова Л. Г., Максимов А. В. Нейрокомпьютеры: учебное пособие для вузов. Москва: МГТУ им. Н. Э. Баумана, 2004. 400 с.