МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Методичні вказівки
до практичних робіт з дисципліни

«ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ»

для студентів усіх форм навчання
спеціальності 121 - Інженерія програмного забезпечення

ЗАТВЕРДЖЕНО
кафедрою «Програмної інженерії».
Протокол № 2 від 25.09.2023р.

ХАРКІВ 2023 р.

Методичні вказівки до практичних робіт з дисципліни «Інтелектуальний аналіз даних» для студентів усіх форм навчання спеціальності 121 – Інженерія програмного забезпечення./ Упоряд.
І.В. Афанасьєва, К.Г. Онищенко – Харків: ХНУРЕ, 2023. – 27 с.

Упорядник Ірина АФАНАСЬЄВА, Костянтин ОНИЩЕНКО

Рецензент: Олег КОБИЛІН, к. т. н, доц. каф. «ІНФ»

ЗМІСТ

ВСТУП

Засоби сучасної інформаційної технології в останній час уможливили накопичення і зберігання великих обсягів даних про бізнесові процеси. Ці дані можуть знаходитися в корпоративних базах або сховищах даних. Вони містять важливі закономірності і зв’язки між системними характеристиками, які можуть бути використані для прийняття обгрунтованих управлінських рішень. Наразі виникла проблема розробки методів відкриття таких закономірностей, про існування яких користувачі можуть і не знати. Проте традиційний аналіз даних передбачує введення даних в стандартні або настроєні користувачем моделі, тобто в будь-якому випадку допускається, що зв'язки між різними показниками добре відомі і можуть бути виражені математично. Однак, в багатьох випадках зв'язки не можуть бути апріорі відомі. У таких ситуаціях моделювання стає неможливим і тут можна застосовувати дейтамайнінг (*Data Mining)* – *інтелектуальний аналіз даних (ІАД).* Тому, особливо важливим аспектом підготовки магістрів за спеціальністю «Програмне забезпечення систем» є успішне засвоєння ними дисципліни "Інтелектуальний аналіз даних”.

Метою практичних занять дисципліни «Інтелектуальний аналіз даних» є вивчення методів первісної обробки даних, видів регресійного аналізу, а саме вивчення методу наіменьших квадратів, вивчення деяких алгоритмів текстовиї класифікації, на прикладі стандартного алгоритму tf-idf, та ознайомити студентів із інтелектуальними агентам, їх властивостями на прикладі розв’язання задач.

Теми відповідають змістовним модулям, що відповідають кредитно-модульному підходу під час викладання та вивчення дисциплін.

Перше практичне заняття присвячено розгляду та розробці у пакеті STATISTICA методу «МНК». На другому занятті розглядаються текстова класифікація, де студенти зможуть розрахувати достовірність текстових даних, та на четвертому занятті студенти розроблятимуть алгоритми інтелектуальних мобільних агентів. Кожне заняття супроводжується питаннями для самоперевірки засвоєння матеріалу та завданнями для самостійного виконання.

1 ПРАКТИЧНЕ ЗАНЯТТЯ № 1. РЕГРЕСІЙНИЙ АНАЛІЗ. МЕТОД «МНК»

1.1 Мета заняття

Метою є вирішення завдяки регресійному аналізу, а саме методу МНК поставлених задач.

1.2 Теоритичні відомості

Пакет прикладних програм «STATISTICA» - являє собою статистичну графічну систему, призначену для вирішення задач математичної та прикладної статистики. Він може бути успішно використаний для ідентифікації статичних характеристик технологічних процесів засобами регресійного та кореляційного аналізів, гребеневими та робастними засобами.

ППП дозволяє заздалегідь вибрати структуру моделі виходячи з статистичного аналізу вхідних даних. Нехай об'єктом дослідження є технологічний процес, який описується статичною залежністю У = F(X) , де X - вектор входу та Y - вектор виходу, F - оператор моделі, що характеризує зв'язок між входом та виходом об'єкта.

При дослідженні процесу спостерігачеві доступні значення вхідних X={xj} та вихідних Y={yi} змінних, що записуються у вигляді таблиці вхідних даних; j=1, p - кількість вхідних змінних; i = 1, n - кількість вимірів.

Внаслідок наявності помилок виміру, які є в експерименті, та впливу різноманітних неврахованих факторів, Y є випадковим процесом із певними статистичними характеристиками.

Кожне значення yt в t-й час є реакція на одночасне вплив xtj, j=1.. p, у той час. Якщо об'єкт характеризується інерцією (затримкою), де yt є реакція на xtj -

t, де t = tx-ty, tx - час початку дії незалежної змінної X, ty - час початку реакції Y на X.

Будь-яка модель відображає лише деякі особливості об'єкта та ніколи не буває його точною копією. Отже, не можна говорити про «справжню» модель у сенсі слова. У окремих випадках під «істинним» значенням розуміють умовне математичне очікування Y при заданих значеннях X, тобто M[Y]=M{y/x}=XB+M[E], де M [.] - символ математичного очікування; E – вектор помилок; B={bl}, l=1, m – вектор параметрів коефіцієнтів моделі.

Структура виразу може бути описана різноманітними функціями, але в цьому випадку обов'язково лінійними щодо коефіцієнтів, bI, I=1, m.

Створимо файл даних, що характеризує виробничо-господарську діяльність машинобудівного підприємства (за прикладом розділу 3). Вихідні дані наведено у таблиці 1.1.

Y1 – продуктивність праці;

Y2 - індекс зниження собівартості продукції;

Y3 – рентабельність;

X4 – трудомісткість одиниці продукції;

X5 - питома вага робітників у складі ППП;

X6 - питома вага покупних виробів;

X7 – коефіцієнт змінності обладнання;

X8 - премії та винагороди на одного працівника;

X9 - питома вага втрат від шлюбу;

X10 - фондовіддача;

X11 – середньорічна чисельність ППП;

X12 – середньорічна вартість ОПФ;

X13 – середньорічний фонд заробітної плати ППП;

X14 – фондоозброєність праці;

X15 - оборотність нормованих оборотних коштів;

X16 - оборотність ненормованих оборотних засобів;

X17 – невиробничі витрати;

Таблиця 1.1 – Вихідні дані

№ під-ємства	Y1	Y2	Y3	X4	X5	X6	X7	X8	X9	X10
1	9,26	204,20	13,26	0,23	0,78	0.40	1.37	1,23	0.23	1.45
2	9,38	209,60	10,16	0,24	0,75	0.26	1.49	1,04	0.39	1.30
3	12,11	222,60	13,72	0,19	0,68	0.40	1.44	1,80	0.43	1.37
4	10,81	236,70	12,85	0,17	0,70	0.50	1.42	0.43	0.18	1.65
5	9,35	62,00	10,63	0,23	0,62	0.40	1.35	0,88	0.15	1.91
6	9,87	53,10	9,12	0,43	0,76	0.19	1.39	0,57	0.34	1.68
7	8,17	172,10	25,83	0,31	0,73	0.25	1.16	1,72	0.38	1.94
8	9,12	56,50	23,39	0,26	0,71	0.44	1.27	1,70	0.09	1.89
9	5,88	52,60	14,68	0,49	0,69	0.17	1.16	0.84	0.14	1.94
10	6,30	46,60	10,05	0,36	0,73	0.39	1.25	0,60	0.21	2.06
11	6,22	53,20	13,99	0,37	0,68	0.33	1.13	0,82	0.42	1.96
12	5,49	30,10	9,68	0,43	0,74	0.25	1.10	0,84	0.05	1.02
13	6,50	146,40	10,03	0,35	0,66	0.32	1.15	0,67	0.29	1.85
14	6,61	18,10	9,13	0,38	0,72	0.02	1.23	1,04	0.48	0.88
15	4,32	13,60	5,37	0,42	0,68	0.06	1.39	0,66	0.41	0.62
16	7,37	89,80	9,86	0,30	0,77	0.15	1.38	0,86	0.62	1.09
17	7,02	62,50	12,62	0,32	0,78	0.08	1.35	0,79	0.56	1.60
18	8,25	46,30	5,02	0,25	0,78	0.20	1.42	0,34	1.76	1.53
19	8,15	103,50	21,18	0,31	0,81	0.20	1.37	1,60	1.31	1.40
20	8,72	73,30	25,17	0,26	0,79	0.30	1.41	1,46	0.45	2.22
21	6,64	76,60	19,40	0,37	0,77	0.24	1.35	1,27	0.50	1.32
22	8,10	73,01	21,0	0,29	0,78	0.10	1.48	1,58	0.77	1.48

23	5,52	32,30	6,57	0,34	0,72	0.11	1.24	0,68	1.20	0.68
24	9,37	199,60	14,19	0,23	0,79	0.47	1.40	0,86	0.21	2.30
25	13,17	598,10	15,81	0,17	0,77	0.53	1.45	1,98	0.25	1.37
26	6,67	71,20	5,23	0,29	0,80	0.34	1.40	0,33	0.15	1.51
27	5,68	90,80	7,99	0,41	0,71	0.20	1.28	0,45	0.66	1.43
28	5,22	82,10	17,50	0,41	0,79	0.24	1.33	0,74	0.74	1.82
29	10,02	76,20	17,16	0,22	0,76	0.54	1.22	0,03	0.32	2.62
30	8,16	119,50	14,54	0,29	0,78	0.40	1,28	0,99		
31	3,78	21,90	6,24	0,51	0,62	0.20	1,47	0,24		
32	6,48	48,40	12,08	0,36	0,75	0.64	1,27	0,57		
33	10,44	173,50	9,49	0,23	0,71	0.42	1,51	1,22		
34	7,65	74,10	9,28	0,26	0,74	0.27	1,46	0,68		
35	8,77	68,60	11,42	0,27	0,65	0.37	1,27	1,00		
36	7,00	60,80	10,31	0,29	0,66	0.38	1,43	0,81		
37	11,06	355,60	8,65	0,01	0,84	0.35	1,50	1,27		
38	9,02	264,80	10,94	0,02	0,74	0.42	1,35	1,14		
39	13,28	526,60	9,87	0,18	0,75	0.32	1,41	1,89		
40	9,27	118,60	6,14	0,25	0,75	0.33	1,47	0,67		
41	6,70	37,10	12,93	0,31	0,79	0.29	1,35	0,96		
42	6,69	57,70	9,78	0,38	0,62	0.30	1,40	0,67		
43	9,42	51,60	13,22	0,24	0,70	0.56	1,20	0,98		
44	7,24	64,70	17,29	0,31	0,66	0.42	1,15	1,16		
45	5,39	48,30	7,11	0,42	0,69	0.26	1,09	0,54		

46	5,61	15,00	22,49	0,51	0,71	0.16	1,26	1,23
47	5,59	87,50	12,14	0,31	0,73	0.45	1,36	0,78
48	6,57	108,40	15,25	0,37	0,65	0.31	1,15	1,16
49	6,54	267,30	31,34	0,16	0,82	0.08	1,87	4,44
50	4,23	34,20	11,56	0,18	0,80	0.68	1,17	1,06
51	5.22	26,80	30,14	0,43	0,83	0.03	1,61	2,13
52	18.00	43.6	19.71	0.40	0.70	0.02	1.34	1.21
53	11.03	72.02	23.56	0.31	0.74	0.22	1.22	2.20

№ під-ємства	**X11**	**X12**	**X13**	**X14**	**X15**	**X16**	**X17**
1	26006	167.69	47750	6.40	72.00	8.64	15.06
2	23935	186.10	50391	7.80	97.20	9.00	20.09
3	22589	220.45	43149	9.76	80.28	14.76	15.98
4	21220	169.30	41089	7.90	51.48	10.08	18.27
5	7394	39.53	14257	5.35	105.12	14.76	14.42
6	11586	40.41	22661	9.90	128.52	10.44	22.76
7	26609	102.96	52509	4.50	94.68	14.76	15.41
8	7801	37.02	14903	4.88	85.32	20.52	19.35
9	11587	45.74	25587	3.46	76.32	14.40	16.83
10	9475	40.07	16821	3.60	153.00	24.84	30.53
11	10811	45.44	19459	3.56	107.64	11.16	17.98
12	6371	41.08	12973	5.65	90.72	6.48	22.09
13	26761	136.14	50907	4.28	82.44	9.72	18.29
14	4210	42.39	6920	8.85	79.92	3.24	26.05
15	3557	37.39	5736	8.52	120.96	6.48	26.20

16	14148	101.78	26705	7.19	84.60	5.40	17.26
17	9872	47.55	20068	4.82	85.32	6.12	18.83
18	5975	32.61	11487	5.46	101.52	8.64	19.70
19	16662	103.25	32029	6.20	107.64	11.88	16.87
20	9166	38.95	18946	4.25	85.32	7.92	14.63
21	15118	81.32	28025	5.38	131.76	10.08	22.17
22	11429	67.26	20968	5.88	116.64	18.72	22.62
23	6462	59.92	11049	9.27	138.24	13.68	26.44
24	24628	107.34	45893	4.36	156.96	16.56	22.26
25	49727	512.60	99400	10.31	137.52	14.76	19.13
26	11470	53.81	20719	4.69	135.72	7.92	18.28
27	19448	80.83	36813	4.16	155.52	18.36	28.23
28	18963	59.42	33956	3.13	48.60	8.28	12.39
29	9185	36.96	17016	4.02	42.84	14.04	11.64
30	17478	91.43	34873	5.23	142.20	16.92	8.62
31	6265	17.16	11237	2.74	145.80	11.16	20.10
32	8810	27.29	17306	3.10	120.52	14.76	19.41
33	17659	184.33	39250	10.44			
34	10342	58.42	19074	5.65			
35	8901	59.40	18452	6.67			
36	8402	49.63	17500	5.91			
37	32625	391.27	7888	11.99			
38	31160	258.62	58947	8.30			

39	46461	75.66	94697	1.63
40	13833	123.68	29626	8.94
41	6391	37.21	11688	5.82
42	11115	53.37	21955	4.80
43	6555	32.87	12243	5.01
44	11085	45.63	20193	4.12
45	9484	48.41	20122	5.10
46	3967	13.58	7612	3.49
47	15283	63.99	27404	4.19
48	20874	104.55	39648	5.01
49	19418	222.11	43799	11.44
50	3351	25.76	6235	7.67
51	6338	29.52	11524	4.66
52	9756	41.99	17309	4.30
53	11795	78.11	22225	6.62

Під час вирішення завдань побудови моделі реального об'єкта перелічені припущення класичного РА є апріорними заданими вимогами до властивостей об'єкта. Проте насправді об'єкт може мати заданими властивостями. У зв'язку з цим при використанні РА для обробки експериментальної інформації необхідно вирішити низку статистичних проблем:

1) провести статистичний аналіз експериментальних даних;

2) задати структуру моделі та отримати найкращі точкові та інтервальні оцінки параметрів bi, I=1, m;

3) провести інтерпретацію моделі, перевіривши гіпотези щодо її параметрів, оцінивши адекватність моделі та перевіривши припущення, на які засновані РА, використовуючи аналіз залишків.

1.3 Порядок виконання роботи

Розглянемо докладніше суттєвість перелічених вище статистичних проблем.

Статистичний аналіз експериментальних даних проводиться з метою визначення класу об'єкта, вибору структури та засоби оцінки параметрів регресійної залежності. Результати аналізу є апостеріорною інформацією для формалізованого вибору засобу ідентифікації.

Статистичний аналіз експериментальних даних з використанням ППП «STATISTICA» включає побудову двовимірних діаграм розсіювання, обчислення одновимірних статистик, побудову гістограм, графіків функцій розподілу, побудови довірчих інтервалів для математичного очікування та дисперсії, обчислення кореляційної функції. Перевірка гіпотез стаціонарності та випадковості вибірки даних, на жаль, не включена до ППП STATISTICA.

Вигляд моделі вибирається виходячи з виду діаграм розсіювання вхідних та вихідних параметрів. Нелінійні моделі приводяться до лінійних шляхом логарифмічного перетворення або заміни змінної.

Як приклад розглянемо набір даних, що описують процес виробництва.

Залежність між параметрами подаємо у вигляді

$$Y = F(X1, X2, ..., B, E),$$

де Y – продуктивність праці.

При розгляді залежності Y = F (X1, X2, ..., B, E), Y є випадковою величиною, значення X можуть розглядатися або як випадкові або як фіксовані. При випадкових значеннях очікуване значення вихідної змінної Y розглядається як умовне математичне очікування $E\{Y/X\}$. У цьому випадку необхідно перевірити гіпотезу про випадковість вхідних та вихідних змінних.

1.4 Зміст звіту

Кожний член бригади оформляє звіт, у якому наводить:

1) цілі і задачі практичної роботи;

2) призначення і коротку характеристику використовуваних програм;

3) розпечатування отриманих результатів;

4) висновки про зроблену роботу.

2 ПРАКТИЧНЕ ЗАНЯТТЯ №2. ТЕКСТОВА КЛАСИФІКАЦІЯ

2.1 Мета заняття

Ознайомитися з особливостями текстової класифікації та виконати завдання за допомогою алгоритму tf-idf.

2.2 Теоритичні відомості

Класична векторна модель простору

Опис, переваги та обмеження класичної векторної просторової моделі

Глобальна інформація

На відміну від моделі підрахунку термінів, модель векторного простору Солтона [1] включає локальну та глобальну інформацію

Рівняння 1: Вага члена

$$w_i = tf_i * log\left(\frac{D}{df_i}\right)$$

де tfi – частота термінів (підраховується термін) або кількість разів, коли термін і зустрічається в документі. Це стосується місцевої інформації.

dfi – частота документа або кількість документів, що містять термін і

D – кількість документів у базі даних.

Співвідношення dfi/D – це ймовірність вибору документа, що містить запитуваний термін із колекції документів. Це можна розглядати як глобальну ймовірність для всієї колекції. Таким чином, термін log(D/dfi) є зворотною частотою документа, IDFi та обліковує глобальну інформацію. Наступний малюнок ілюструє співвідношення між локальними та глобальними частотами в

ідеальній колекції бази даних, що складається з п’яти документів D1, D2, D3, D4 та D5. Лише три документи містять термін «ЦАР». Запит системи для цього терміна дає значення IDF log(5/3) = 0,2218.

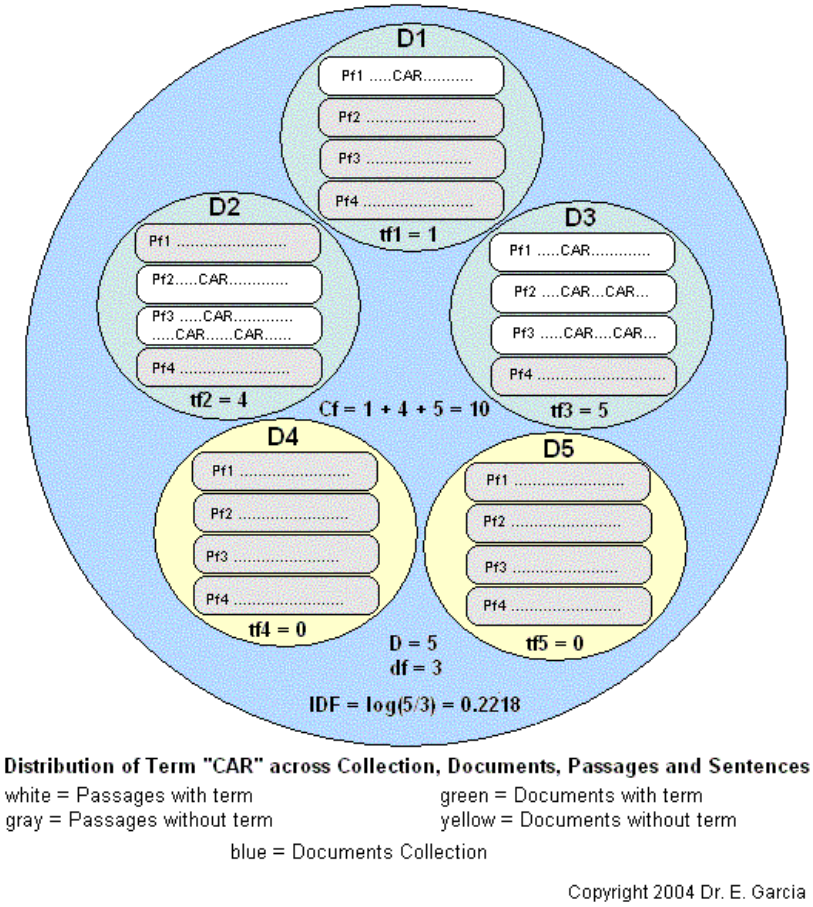

Рисунок 2.1 – Векторна модель простору

Елементи самоподібності

Ті з нас, хто спеціалізується на прикладній фрактальній геометрії, визнають самоподібну природу цієї фігури в деяких масштабах. Зауважте, що збірники складаються з документів, документи складаються з уривків, а уривки складаються з речень. Таким чином, для терміна і в документі j ми можемо говорити в термінах частоти збирання (Cf), частоти терміну (tf), частоти проходження (Pf) і частоти речень (Sf)

Eq 2(a, b, c): $$Cf_i = \sum_j tf_{i,j}$$

$$tf_{i,j} = \sum_p Pf_{i,j,p}$$

$$Pf_{i,j,p} = \sum_s Sf_{i,j,p,s}$$

Eq 2(b) неявно міститься в Eq 1. Моделі, які намагаються пов'язати ваги термінів із значеннями частоти, повинні брати до уваги характер масштабування релевантності. Звичайно, так зване співвідношення «щільності ключових слів», яке просувається багатьма оптимізаторами пошукових систем (SEO), не відноситься до цієї категорії.

3.3 Порядок виконання роботи

Щоб зрозуміти рівняння 1, використаємо тривіальний приклад. Щоб спростити, припустімо, що ми маємо справу з базовою векторною моделлю термів, у якій ми

1. не враховують, ДЕ в документах зустрічаються терміни.

2. використовувати всі терміни, включаючи дуже поширені терміни та стоп-слова.

3. не зводити терміни до кореневих (основні).

4. використовувати необроблені частоти для термінів і запитів (ненормалізовані дані).

Я надаю наступний приклад, наданий професорами Девідом Гроссманом і Офіром Фрідером з Іллінойського технологічного інституту [2]. Це один із найкращих прикладів обчислення вектора термів, доступних онлайн.

• До речі, д-р Гроссман і д-р Фрідер є авторами авторитетної книги «Інформаційний пошук: алгоритми та евристика». Спочатку опублікований у 1997 році, нове видання доступне зараз через Amazon.com [3]. Це обов'язкова література для аспірантів, пошукових інженерів і маркетологів пошукових

систем. У книзі йдеться про те, що стоїть за ІЧ-системами та пошуковими алгоритмами.

Припустимо, ми надсилаємо ІЧ-системі запит на запит "золото срібло вантажівка". Колекція бази даних складається з трьох документів (D = 3) з таким вмістом

D1: «Відправка золота, пошкоджена пожежею»

D2: «Доставка срібла прибула у срібній вантажівці»

D3: «Поставка золота прибула вантажівкою»

Результати пошуку зведені таблицю, що наведена на рис. 2.2.

TERM VECTOR MODEL BASED ON $w_i = tf_i * IDF_i$

Query, Q: "gold silver truck"
D_1: "Shipment of gold damaged in a fire"
D_2: "Delivery of silver arrived in a silver truck"
D_3: "Shipment of gold arrived in a truck"
D = 3; IDF = log(D/df_i)

		Counts, tf_i						Weights, $w_i = tf_i * IDF_i$			
Terms	**Q**	**D_1**	**D_2**	**D_3**	**df_i**	**D/df_i**	**IDF_i**	**Q**	**D_1**	**D_2**	**D_3**
a	0	1	1	1	3	3/3 = 1	0	0	0	0	0
arrived	0	0	1	1	2	3/2 = 1.5	0.1761	0	0	0.1761	0.1761
damaged	0	1	0	0	1	3/1 = 3	0.4771	0	0.4771	0	0
delivery	0	0	1	0	1	3/1 = 3	0.4771	0	0	0.4771	0
fire	0	1	0	0	1	3/1 = 3	0.4771	0	0.4771	0	0
gold	1	1	0	1	2	3/2 = 1.5	0.1761	0.1761	0.1761	0	0.1761
in	0	1	1	1	3	3/3 = 1	0	0	0	0	0
of	0	1	1	1	3	3/3 = 1	0	0	0	0	0
silver	1	0	2	0	1	3/1 = 3	0.4771	0.4771	0	0.9542	0
shipment	0	1	0	1	2	3/2 = 1.5	0.1761	0	0.1761	0	0.1761
truck	1	0	1	1	2	3/2 = 1.5	0.1761	0.1761	0	0.1761	0.1761

Рисунок 2.2 – Результати пошуку

Табличні дані базуються на прикладі доктора Гроссмана. Я додав останні чотири стовпці, щоб проілюструвати всі обчислення ваги терміну. Давайте проаналізуємо необроблені дані стовпчик за стовпцем.

1. Стовпці 1–5: спочатку ми створюємо індекс термінів із документів і визначаємо кількість термінів tfi для запиту та кожного документа Dj.

2. Стовпці 6–8: по-друге, ми обчислюємо частоту di для кожного документа. Оскільки IDFi = log(D/dfi) і D = 3, цей розрахунок є простим.

3. Стовпці 9–12: По-третє, ми беремо добутки tf*IDF і обчислюємо ваги членів. Ці стовпці можна розглядати як розріджену матрицю, в якій більшість записів дорівнюють нулю.

Тепер ми розглядаємо ваги як координати у векторному просторі, фактично представляючи документи та запит як вектори. Щоб дізнатися, який вектор документа ближчий до вектора запиту, ми використовуємо аналіз подібності, представлений у частині 2.

Аналіз подібності

Спочатку для кожного документа та запиту ми обчислюємо всі довжини векторів (нульові члени ігноруються):

$$|D_1| = \sqrt{0.4771^2 + 0.4771^2 + 0.1761^2 + 0.1761^2} = \sqrt{0.5173} = 0.7192$$
$$|D_2| = \sqrt{0.1761^2 + 0.4771^2 + 0.9542^2 + 0.1761^2} = \sqrt{1.2001} = 1.0955$$
$$|D_3| = \sqrt{0.1761^2 + 0.1761^2 + 0.1761^2 + 0.1761^2} = \sqrt{0.1240} = 0.3522$$

$$\therefore |D_i| = \sqrt{\sum_i w_{i,j}^2}$$

$$|Q| = \sqrt{0.1761^2 + 0.4771^2 + 0.1761^2} = \sqrt{0.2896} = 0.5382$$

$$\therefore |Q| = \sqrt{\sum_i w_{Q,j}^2}$$

Далі ми обчислюємо всі скалярні добутки (нульові добутки ігноруємо):

$$Q \bullet D_1 = 0.1761 * 0.1761 = 0.0310$$
$$Q \bullet D_2 = 0.4771 * 0.9542 + 0.1761 * 0.1761 = 0.4862$$
$$Q \bullet D_3 = 0.1761 * 0.1761 + 0.1761 * 0.1761 = 0.0620$$

$$\therefore Q \bullet D_i = \sum_i w_{Q,j} w_{i,j}$$

Тепер розрахуємо значення подібності:

$$\text{Cosine } \theta_{D_1} = \frac{Q \bullet D_1}{|Q| * |D_1|} = \frac{0.0310}{0.5382 * 0.7192} = 0.0801$$

$$\text{Cosine } \theta_{D_2} = \frac{Q \bullet D_2}{|Q| * |D_2|} = \frac{0.4862}{0.5382 * 1.0955} = 0.8246$$

$$\text{Cosine } \theta_{D_3} = \frac{Q \bullet D_3}{|Q| * |D_3|} = \frac{0.0620}{0.5382 * 0.3522} = 0.3271$$

$$\therefore \text{Cosine } \theta_{D_i} = \text{Sim}(Q, D_i)$$

$$\therefore \text{Sim}(Q, D_i) = \frac{\sum_i w_{Q,j} w_{i,j}}{\sqrt{\sum_j w_{Q,j}^2} \sqrt{\sum_i w_{i,j}^2}}$$

Нарешті ми сортуємо та ранжуємо документи в порядку спадання відповідно до значень подібності:

Rank 1: Doc 2 = 0.8246

Rank 2: Doc 3 = 0.3271

Rank 3: Doc 1 = 0.0801

Спостереження

Цей приклад ілюструє кілька фактів. По-перше, дуже часті терміни, такі як "a", "in" і "of", як правило, отримують низьку вагу - значення нуль у цьому випадку. Таким чином, модель правильно передбачає, що дуже загальні терміни, які зустрічаються в багатьох документах у колекції, не є хорошими дискримінаторами релевантності. Зверніть увагу, що це міркування базується на глобальній інформації; тобто термін IDF. Саме тому ця модель є кращою, ніж модель підрахунку термінів, розглянута в частині 2. По-третє, замість обчислення довжин окремих векторів і скалярних добутків ми можемо заощадити обчислювальний час, застосувавши безпосередньо функцію подібності

Eq 3: $$Sim(Q,D_i) = \frac{\sum_i w_{Q,j} w_{i,j}}{\sqrt{\sum_j w_{Q,j}^2} \sqrt{\sum_i w_{i,j}^2}}$$

Звичайно, нам все ще потрібно знати індивідуальні значення tf і IDF.

Обмеження моделі

Як базова модель, обговорювана термінова векторна схема має кілька обмежень. По-перше, це дуже інтенсивне обчислення. З обчислювальної точки зору це дуже повільно, вимагаючи багато часу обробки. По-друге, кожного разу, коли ми додаємо новий термін у простір термінів, нам потрібно перераховувати всі вектори. Як зазначили ЛІ, ЧУАНГ і СІМОНС [4], обчислення довжини вектора запиту (перший член у знаменнику рівняння 3) вимагає доступу до кожного терміну документа, а не лише до термінів, указаних у запиті.

Інші обмеження включають

1. Довгі документи: дуже довгі документи ускладнюють вимірювання подібності (вектори з малими скалярними добутками та високою розмірністю)
2. Помилкові негативні збіги: документи з подібним вмістом, але різними словниками, можуть призвести до поганого внутрішнього продукту. Це обмеження інфрачервоних систем, керованих ключовими словами.
3. Помилкові збіги: неправильне формулювання, видалення префікса/суфікса або синтаксичний аналіз може призвести до помилкових звернень (падання, падіння + ing; терапевт, + ґвалтівник, + rap + ist; Marching, March + ing; GARCIA, GAR + CIA). Це лише обмеження попередньої обробки, а не зовсім обмеження векторної моделі.
4. Семантичний вміст: системам для обробки семантичного вмісту може знадобитися використання спеціальних тегів (контейнерів).

Ми можемо вдосконалити модель:

1. отримання набору ключових слів, що представляють кожен документ.

2. видалення всіх стоп-слов і дуже поширених термінів ("a", "in", "of" тощо).

3. похідні терміни до їх коренів.

4. обмеження векторного простору іменниками та декількома описовими прикметниками та дієсловами.

5. використання маленьких файлів підписів або не надто великих інвертованих файлів.

6. використання прийомів тематичного відображення.

7. обчислення підвекторів (векторів переходів) у довгих документах

8. не витягувати документи нижче визначеного косинусного порогу

Про полісемію та синонімію

Основним недоліком цієї та всіх векторних моделей термінів є те, що терміни вважаються незалежними (тобто між термінами не існує зв’язку). Часто це не так. Терміни можуть бути пов’язані між собою

1. Багатозначність; тобто терміни можна використовувати для вираження різних речей у різних контекстах (наприклад, водіння автомобіля та результати водіння). Таким чином, деякі нерелевантні документи можуть мати велику схожість, оскільки вони можуть використовувати деякі слова з запиту. Це впливає на точність.

2. Синонімічність; тобто терміни можна використовувати для вираження того самого (наприклад, страхування автомобіля та автострахування). Таким чином, схожість деяких відповідних документів із запитом може бути низькою лише через те, що вони не мають однакових термінів. Це впливає на запам'ятовування.

З цих двох синонімічність може негативно вплинути на показники вектора термінів.

2.3 Порядок виконання роботи

1. Виконати індивідуальне завдання за номером.
2. Оформити звіт з роботи.

Індивідуальне завдання 1:

Vector Space model based on $w_i = tf_i \cdot IDF_i$

Умова:

Query, Q: “silver black car”

D1: “Advantages of good looking of black car”

D2: “The silver has good looking shine”

D3: “Advantages of good looking silver that the silver is shine”

D4: “Both silver and black have good looking shine”

Треба:

1. Для кожного документа порахувати довжину векторів
2. Порахувати запит
3. Порахувати скалярний твір $Q \bullet D_i$
4. Порахувати cosine measure (similarity measure)
5. Сортувати документи за значимістю щодо similarity measure
6. Висновки про отриманий результат.

Індивідуальне завданне 2:

Vector Space model based on $w_i = tf_i \cdot IDF_i$

Умова:

Q: “Data mining”

D1: “Text Classification is filed of Data Mining”

D2: “Data is important for Text Mining”

D3: “Text mining for Text Processing”

Треба:

1. Для кожного документа порахувати довжину векторів
2. Порахувати запит
3. Порахувати скалярний твір $Q \bullet D_i$
4. Порахувати cosine measure (similarity measure)
5. Сортувати документи за значимістю щодо similarity measure
6. Висновки про отриманий результат.

Індивідуальне завдання 3:

Vector Space model based on $w_i = tf_i \cdot IDF_i$

Умова:

Q “Data Mining”

D1: Text classification is filed of data mining

D2: Data is important for text mining

D3: Text mining for text processing

Треба:

1. Для кожного документа порахувати довжину векторів
2. Порахувати запит
3. Порахувати скалярний твір $Q \bullet D_i$
4. Порахувати cosine measure (similarity measure)
5. Сортувати документи за значимістю щодо similarity measure
6. Висновки про отриманий результат.

Індивідуальне завдання 4:

Vector Space model based on $w_i = tf_i \cdot IDF_i$

Умова:

Q “Text Classification”

D1: Text classification is filed of data mining

D2: Data are important for text classification

D3: Text mining for text processing

Треба:

1. Для кожного документа порахувати довжину векторів
2. Порахувати запит
3. Порахувати скалярний твір $Q \bullet D_i$
4. Порахувати cosine measure (similarity measure)
5. Сортувати документи за значимістю щодо similarity measure
6. Висновки про отриманий результат.

Індивідуальне завдання 5:

Vector Space model based on $w_i = tf_i \cdot IDF_i$

Умова:

Q “Text Mining”

D1: Text classification is filed of data mining

D2: Data are important for text classification

D3: Text mining for text processing

Треба:

1. Для кожного документа порахувати довжину векторів
2. Порахувати запит
3. Порахувати скалярний твір $Q \bullet D_i$
4. Порахувати cosine measure (similarity measure)

5. Сортувати документи за значимістю щодо similarity measure

6. Висновки про отриманий результат.

Інші варіанти завдань слід взяти у викладача.

2.4 Зміст звіту

Звіт має містити:

– титульний аркуш;

– мету, варіант і завдання роботи;

– лаконічний опис теоретичних відомостей;

– текст програми, що обов'язково містить коментарі (якщо треба за завданням);

– вхідні та вихідні дані програми (якщо треба за завданням);

– змістовний аналіз отриманих результатів та висновки.

Під час співбесіди студент повинний виявити знання про мету роботи, по теоретичному матеріалу, про методи виконання кожного етапу роботи, по змісту основних розділів розробленого звіту з демонстрацією результатів на конкретних прикладах. Студент повинний вміти правильно аналізувати отримані результати.

МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ ТА РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. La Rocca M. Advanced Algorithms and Data Structures / Marcello La Rocca. – Shelter Island: Manning Publications, 2021. – 737 с.
2. Scappini A. The Art of Data Analysis: Non-Technical Skills for Data Analysts / Alberto Scappini., 2018. – 304 с.
3. Weidman S. Deep Learning from Scratch: Building with Python from First Principles / Seth Weidman., 2019. – 235 с.
4. Tunstall L. Natural Language Processing with Transformers: Building Language Applications with Hugging Face / L. Tunstall, L. Werra, T. Wolf., 2022. – 383 с.
5. F. Provost Data Science For Business: What You Need to Know About Data Mining & Data-Analytic Thinking, 2018. – 313 с. – (1st Edition).
6. Aggarwal C. Linear Algebra and Optimization for Machine Learning / Charu C. Aggarwal. – Yorktown: Springer, 2020. – 1087 с. – (1st Edition).
7. J. Zaki M. Data Mining and Machine Learning: Fundamental Concepts and Algorithms / M. J. Zaki, W. Meira, Jr. – Cambridge: Cambridge University Press, 2020. – 776 с. – (2nd Edition).
8. George A. Python Text Mining: Perform Text Processing, Word Embedding, Text Classification and Machine Translation / Alexandra George., 2022. – 320 с. – (1st Edition).
9. Runkler T. Data Analytics: Models and Algorithms for Intelligent Data Analysis / Thomas A. Runkler. – Munich: Springer, 2020. – 176 с.
10. Xiao P. Artificial Intelligence Programming with Python / Perry Xiao. – New Jersey: John Wiley & Sons, Inc., 2022. – 720 с.
11. Russell S. Artificial Intelligence: A Modern Approach / S. Russell, P. Norvig., 2021. – 1168 с. – (4th Edition).
12. Osinga D. Deep Learning Cookbook: Practical Recipes to Get Started Quickly / Douwe Osinga., 2018. – 251 с. – (1st Edition).

13. Machine Learning for Beginners: The Ultimate Guide to Artificial Intelligence: Matt Henderson, This is Charlotte Press, 2019 – 100 c.

14. Berman J. Principles and Practice of Big Data / Jules J. Berman., 2018. – 480 c. – (2nd Edition).

15. Tunstall L. Natural Language Processing with Transformers: Building Language Applications with Hugging Face / L. Tunstall, L. Werra, T. Wolf., 2022. – 383 c.