Міністерство освіти і науки України
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики

Ministry of Education and Science of Ukraine
Ivan Franko National University of Lviv
Faculty of Applied Mathematics and Informatics

МІЖНАРОДНА СТУДЕНТСЬКА НАУКОВА КОНФЕРЕНЦІЯ З ПРИКЛАДНОЇ МАТЕМАТИКИ ТА КОМП'ЮТЕРНИХ НАУК МСНКПМКН – 2022

INTERNATIONAL STUDENT SCIENTIFIC CONFERENCE ON APPLIED MATHEMATICS AND COMPUTER SCIENCES ISSCAMCS – 2022

Львів, 5-6 травня 2022
Lviv, May 5-6, 2022

Метою конференції є сприяння співпраці між студентами, що вивчають споріднені області прикладної математики та комп'ютерних наук.

Головні теми: інформаційні технології та системи, комп'ютерне моделювання, інженерія.

Організаційний комітет:

Дияк Іван – декан факультету прикладної математики та інформатики.
Горлач Віталій – заступник декана факультету прикладної математики та інформатики.
Лавришин Ірина – голова студентської ради факультету прикладної математики та інформатики.
Шевченко Ірина – студентка факультету прикладної математики та інформатики.
Дмитровська Ярина – студентка факультету прикладної математики та інформатики.

© Факультет прикладної математики та інформатики, 2022

© Faculty of Applied Mathematics and Informatics, 2022

Авторський покажчик Author's index

Abayomi Dennis Epebinu *112*
Anastasiia Bondarenko *50*
Iryna Kovalchyk *99*
Iryna Onyshkevych *71*
Joshua Junior John *117*
Oleksandr Skorupskyy *4*
Vitaliy Horlatch *50, 54*
Volodymyr Milchanovskyi *54*
Yaryna Paslavska *33*
Yuriy Muzychuk *4*
Аліса Трей *90*
Анастасія Петришин *128*
Андрій Бурцьо *45*
Андрій Веліховський *67*
Артем Волобуєв *85*
Богдан-Микола Добосевич *139*
Вікторія Чумакевич *13*
Віталій Горлач *8*
Дана Зарецька *17*
Данило Брозь *110*
Даніель Брецко *75*
Дмитро Альянах *95*
Іван Винявський *29*
Катерина Старовойт *21*
Любов-Василина Вовк *107*
Любомир Фадєєв *41*
Мар'яна Шмуль *124*
Микола Баранов *17*
Микола Ковалик *135*
Михайло Бордун *79*
Олег Іванків *67, 135*
Олег Смаль *83*
Олександр Зелінський *8*
Роман Куновський *25*
Ростислав Байцар *88*
Сергій Іванов *17*
Сергій Тороній *63*
Софія Ковальчук *132*
Софія Яропуд *141*
Станіслав Чернецький *47*
Христина Павліковська *59*
Юліана Юрченко *36*
Юлія Жорняк *102*
Юрій Лебедін *8*
Ярослав Соколовський *36*

INFORMATION RETRIEVAL SYSTEM DEVELOPMENT

Oleksandr Skorupskyy [1,2], Yuriy Muzychuk [1]

[1] Ivan Franko National University of Lviv (Department of Applied Mathematics)

[2] University of L'Aquila (DISIM)

Ukraine, Lviv

oleksandr.skorupskyy@lnu.edu.ua

Abstract

The information retrieval process begins with a query to the system. However, when searching for information, the query is not a precise identifier of a specific object in the collection. Unlike classic SQL queries to a database, information retrieval results may match the query completely or partially, so they are usually ranked according to their degree of correspondence.[1] Stages of the IR system development combined with the content analysis techniques were discussed in this paper and applied to the problem of social network analysis.

Keywords: information retrieval, tokenization, normalization, lemmatization, stemming.

Introduction

The object of research of this study is the information retrieval system.

Information retrieval (IR) is the process of obtaining information resources that are relevant to an information need from a collection of those resources. IR can be based on full-text or other content-based indexing.[2] This is the science of searching for information in a document, searching for documents themselves, and for databases of texts, images, sounds or it can also be searching for the metadata that describes data.

The growth of social networks has led to the formation of a new field of study and new data for research. In recent years many algorithms have been created for natural language processing and information retrieval.

The major task of this research is to provide an orderly and clear pipeline for the IR system. The research is focused on IR system development from the real-time data of the particular social network stream.

Stages of the IR system development process

The research was performed by using different text processing methods, e.g. tokenization, stemming, lemmatization, normalization. The task of tokenization is to separate words from punctuation marks, numbers, complexes of letters and numbers, Internet addresses, nicknames, signs, etc.[3] The tasks of stemming, lemmatization

and normalization are similar, but differ in the depth of data processing and approaches to solving the problem. For example, stemming is the process of reducing a word to its base by discarding auxiliary parts, such as endings or suffixes.

Figure 1. Stemming

Lemmatization, in turn, is a more complex method based on determining the basis of the word.

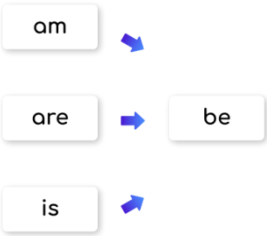


Figure 2. Lemmatization

Text normalization is an even more complex approach to text processing which, for example, takes into account the context and meaning of a particular word.

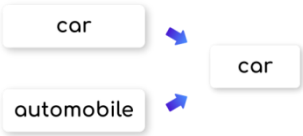


Figure 3. Normalization

Indexing is one of the key principles on which the IR system is based. The indexing procedure allows the transformation of a large set of text data into the data storage model that is easily accessible and fast to process. Another important feature of information retrieval systems is the order of importance of individual words in the

context of the document. The importance of a word can be calculated, for example, by statistically estimating the proportion of the word used in a document and the inversely proportional number of documents in which the word is used throughout the collection.

Results

Different methods of text processing were studied. The information retrieval system was developed based on Twitter data. Figure 4 graphically illustrates stages of the IR system development process. 1 million tweets (January - February 2021) were used for building an index. Content analysis was performed by using the TF-IDF ranking function. “Covid”, “coronavirus”, “vaccine” were the most “important” words.

Figure 4. IR system development process

Discussion

The obtained results of content analysis can be considered as representative and objective due to large size of input data. Taking into account the pandemic situation in the world these results were expected and forecasted.

Conclusions

In the research an IR system was developed based on social network data. Combined with methods of content analysis this system provides the most "important" words of January-February 2021. Future work will focus on the use of different graph algorithms on this data to investigate new features of social networks.

References

1. Manning C., CS276/LING286: Information Retrieval and Web Search / Manning C., Nayak P. // Stanford cource – 2019 [Electronic resourse]. – Available from: https://web.stanford.edu/class/cs276/
2. Manning C. An Introduction to Information Retrieval / Manning C., Raghavan P., Schütze H. // Online edition, Cambridge UP – 2009 – 544 p.
3. Teodorescu M.H. Machine Learning Methods for Strategy Research // HBS Working Paper 18-011. – Harvard Business School – 2017 – 59 p.

РОЗРОБКА СИТЕМИ КЛАСТЕРИЗАЦІЇ АНТИТІЛ НА ОСНОВІ КОЕФІЦІЄНТУ ПЕРЕХРЕСНОГО ЗВ'ЯЗУВАННЯ

Олександр Зелінський, Віталій Горлач, Юрій Лебедін

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

Oleksandr.Zelinskyi@lnu.edu.ua

В умовах пандемії надзвичайно важливими є дослідження які несуть безпосередню користь для виявлення, запобігання та лікування вірусних захворювань, а якщо точніше вірусу Covid-19 або SARS-CoV-2. А в умовах поширення комп'ютерів та іншої потужної обчислювальної техніки зручним та важливим є використання комп'ютерних алгоритмів для виконання завдань пов'язаних з дослідженнями вірусів.

Дано молекулу вірусу SARS-CoV-2, до якої приєднуються два антитіла, для того щоб можна було їх відрізнити одне з них помічається *. Для простоти вважатимемо, що все відбувається на площині, а антитіла це два круги однакового розміру, що приєднуються до меншого круга який представляє молекулу вірусу.

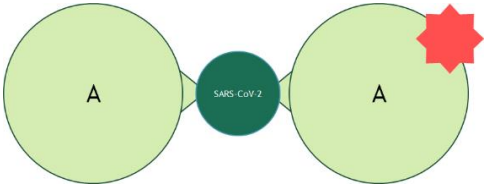


Рис. 1. Модель приєднання антитіл до вірусної молекули

Задача полягає у знаходженні двох антитіл (можуть бути однаковими), таких що знаходяться на оптимальній відстані одне від одного (не перетинаються, не знаходяться занадто близько). Зручним теоретичним способом для цього є розбиття списку антитіл на групи (антитіла з різних груп взаємодіють краще ніж з однієї). Далі буде наведено опис алгоритму розбиття.

Дані з експерименту подані у вигляді таблиці, де кожна комірка це коефіцієнт перехресного зв'язування міченого антитіла (зі стовпця) та не міченого (з рядка). В рядку позначеному як "blank" надані максимальні значення коефіцієнтів перехресного зв'язування для відповідного міченого антитіла.

Labelled	NP1501*	NP1502*	NP1503*	NP1508*	NP1510*	NP1514*	NP1516*	NP1517*
blank	1.089	1.067	1.3664	1.412	1.67	1.07	1.1704	1.11
NP1501	0.449	0.715	1.0664	1.0248	1.136	0.26	0.4172	0.268
NP1502	0.893	0.425	0.336	0.196	0.349	0.3625	0.665	0.45
NP1503	0.768	0.309	0.068	0.052	0.095	0.305	0.7126	0.408
NP1507	0.422	0.856	0.7216	0.6088	0.785	0.1825	0.4256	0.154
NP1508	0.732	0.388	0.1456	0.0848	0.19	0.345	0.8456	0.411
NP1510	0.781	0.382	0.2152	0.1056	0.233	0.495	0.7826	0.527
NP1512	0.79	0.789	0.876	0.7504	0.979	0.735	1.015	0.737
NP1514	0.448	0.822	1.0968	1.0088	1.189	0.2475	0.4858	0.253
NP1516	0.385	1.034	0.9832	0.9304	1.053	0.1775	0.3052	0.152
NP1517	0.425	0.644	0.7952	0.7304	0.885	0.155	0.2758	0.079

Рис. 2. Фрагмент початкових даних

Для подальшої роботи з даними їх позначають за наступним алгоритмом:

1. Цифрою 3 (темно-зеленим кольором) антитіла з поганим зв'язуванням, якщо
$$\frac{-(cell[i][j]-blank[j])}{blank[j]} > 0.75 \quad (1)$$
2. Цифрою 2 (світло-зеленим кольором) антитіла з середнім зв'язуванням, якщо
$$0.5 < \frac{-(cell[i][j]-blank[j])}{blank[j]} \leq 0.75 \quad (2)$$
3. Цифрою 1 (білим кольором) антитіла з хорошим зв'язування, у всіх інших випадках.

label	NP1501*	NP1502*	NP1503*	NP1508*	NP1510*	NP1514*	NP1516*	NP1517*
NP1501	2	1	1	1	1	3	2	3
NP1502	1	2	3	3	3	2	1	2
NP1503	1	2	3	3	3	2	1	2
NP1507	2	1	1	2	2	3	2	3
NP1508	1	2	3	3	3	2	1	2
NP1510	1	2	3	3	3	2	1	2
NP1512	1	1	1	1	1	1	1	1
NP1514	2	1	1	1	1	3	2	3
NP1516	2	1	1	1	1	3	2	3
NP1517	2	1	1	1	1	3	3	3

Рис. 3. Фрагмент позначених даних

Тепер задача полягає у розбитті матриці перехресного зв'язування антитіл на групи за ознакою подібності раніше створеного показника зв'язування для полегшення виявлення оптимальних пар та приблизної локалізації місця зв'язування. Для цього використовують методи кластеризації, а саме k-modes.

k-modes – це алгоритм, який базується на алгоритмі k-means і використовується для кластеризації даних на основі якісних змінних. k-modes визначає кластери на основі відповідності категорій між точками даних. В

даному алгоритмі відстань між двома точками даних X та Y описується як сума не схожих елементів:

$$d_1(X,Y) = \sum_{i=1}^{n} \delta(x_i, y_i)\text{, де } \delta(x_i, y_i) = \begin{cases} 0, & x_i = y_i \\ 1, & x_i \neq y_i \end{cases} \quad (3)$$

Для визначення оптимальної кількості кластерів використовується elbow метод, який для різних значень k буде вибирати значення k у тій точці, де значення істотно не зменшується зі збільшенням значення k.

Для обробки даних та кластеризації використовувалась бібліотека kmodes, pandas, matplotlib та kneed з Python. В результаті виконання програми отримано оптимальне розбиття на 9 кластерів.

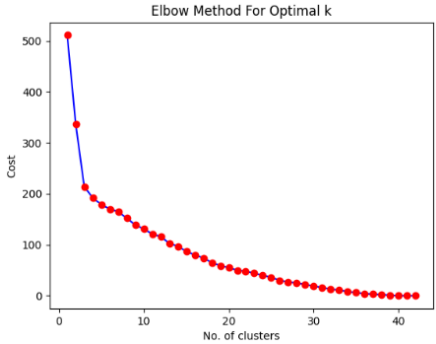


Рис. 4. Фрагмент позначених даних

Antigen by groups				
Group I	Group II	Group III	Group IV	Group V
NP1501, NP1512, NP1514, NP1516, NP1517, NP1521, X211, X215, X218, X220, X275	NP1528	NP1502, NP1503, NP1508, NP1510, NP1520, NP1522, NP1525, NP3708, X190, X221, X271	NP1524, NP3715, X212, X217, X223, X224	NP1526, NP3701, X200, X201
Group VI	Group VII	Group VIII	Group IX	
NP1507	X155, X202, X213, X233, X32, X41	NP1518	NP1527, NP3706	

Рис. 5. Результат розбиття на кластери

Табл. 1. Очікуваний результат розбиття

Групи	Під групи	Елементи
1	A	NP1501, NP1514, NP1516, NP1517, NP1507
	B	X190, NP1526, X200, X201
1B/2		NP1512, NP1521

2		NP1502, NP1503, NP1508, NP1510, NP1520, NP1522, NP1525, X221, X271, NP3701, NP3708
2B/3		NP1528
3	A	X202, X218, NP1518, NP1527
	B	X32, X155, X41, X212, X213, X217, X223, X224, X233, NP1524, NP3715
4	A	NP3706
	B	X211
	C	X215
5		X220
6		X275

З результатів видно, що група I майже відповідає групі 1A в об’єднанні з 1B/2 та 4B, 4C, 5 і 6. Група III майже відповідає групі 2, Група II відповідає групі 2B/3. Група 3B відповідає групі VII та групі IV. Зважаючи на те що кількість елементів які мають бути в однакових групах 30 з 43, то можна вважати, що похибка становить близько 30%.

На рисунках 6 та 7 наведена візуалізація реального та очікуваного розбиття на групи за допомогою кольорів.

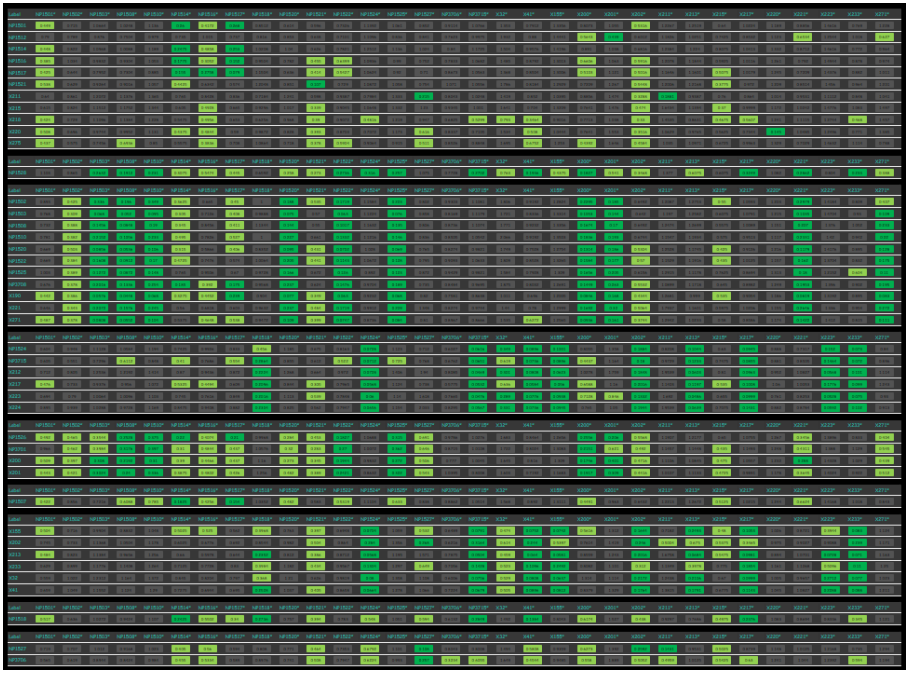


Рис. 6. Візуалізація результатів розбиття

Рис. 7. Візуалізація очікуваного результату розбиття

Список літератури

1. Satyam Kumar Clustering Algorithm for data with mixed Categorical and Numerical features [Electronic resource]. – 2021. – URL: https://towardsdatascience.com/clustering-algorithm-for-data-with-mixed-categorical-and-numerical-features-d4e3a48066a0
2. Z. Huang. Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values (1998). Data Mining and Knowledge Discovery. 2(3): 283–304.
3. Python: K-modes explanation [Electronic resource]. – 2017. – URL : https://stackoverflow.com/questions/42639824/python-k-modes-explanation

ДОСЛІДЖЕННЯ ВИКОРИСТАННЯ РІЗНИХ ДВОВИМІРНИХ ТРИКУТНИХ ЕЛЕМЕНТІВ ДЛЯ МОДЕЛЮВАННЯ СЕРЕДОВИЩ МЕТОДОМ СКІНЧЕННИХ ЕЛЕМЕНТІВ

Вікторія Чумакевич

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

victoriachu21@gmail.com

Постанова плоскої задачі теорії пружності

Маємо O_{xy} - площина деформації. Нехай u, v, w переміщення вздовж осей O_x, O_y та O_z – відповідно. Тоді $u = u(x, y)$, $v = v(x, y)$, $w = 0$. (1)

Співвідношення закону Гука для ізотропного випадку має вигляд:

$$\sigma_{xx} = D\left(\varepsilon_{xx} + \frac{\nu}{1-\nu}\varepsilon_{yy}\right); \quad \sigma_{xy} = D\frac{1-2\nu}{2(1-\nu)}\gamma_{xy};$$
$$\sigma_{yy} = \left(\frac{\nu}{1-\nu}\varepsilon_{xx} + \varepsilon_{yy}\right); \quad \sigma_{zx} = \sigma_{zy} = 0, \tag{2}$$

де E - модуль пружності (модуль Юнга); ν - коефіцієнт Пуасона;

$$D = \frac{(1-\nu)E}{(1+\nu)(1-2\nu)}. \tag{3}$$

Приймемо псевдовектори деформацій та напружень:

$$[\varepsilon] = \left[\varepsilon_{xx}, \varepsilon_{yy}, \gamma_{xy}\right]^T; \quad [\sigma] = \left[\sigma_{xx}, \sigma_{yy}, \sigma_{xy}\right]^T, \tag{4}$$

Використовуючи співвідношення Коші запишемо закон Гука:

$$[\sigma] = [D][\varepsilon]; \quad [D] = D\begin{bmatrix} 1 & \frac{\upsilon}{1-\upsilon} & 0 \\ \frac{\upsilon}{1-\upsilon} & 1 & 0 \\ 0 & 0 & \frac{1-2\upsilon}{2(1-\upsilon)} \end{bmatrix} \tag{5}$$

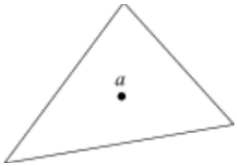
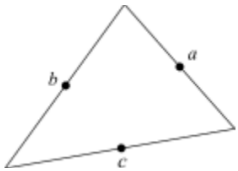
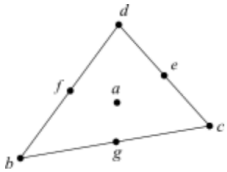
Будемо вважати, що масові сили відсутні. Вектор зовнішнього навантаження приймемо $[p] = \left[p_x, p_y\right]^T$. Функціонал Лагранжа в матричних позначеннях запишеться так:

$$J([U]) = \frac{1}{2}\int_{\Omega}[\varepsilon]^T[D][\varepsilon]d\Omega - \int_{S_\sigma}[p]^T[U]dS \quad . \qquad (6)$$

Використання різних двовимірних елементів

Як відомо скінченими елементами є прості області, такі як лінії, трикутники, прямокутники, піраміди, призми. В своїх дослідженнях я зосередилася на трикутних скінчених елементах. Мною були розглянуті декілька їх видів, а саме лінійні та квадратичні. Дня наглядності наведемо табл. 1, що демонструє їх вузли Гауса разом з ваговими коефіцієнтами.

Табл. 1. Вузли Гауса та вагові коефіцієнти

	Похибка	Точки	Координати L_1	L_2	L_3	Вагові коефіцієнти
	$R = O(h^2)$	a	1/3	1/3	1/3	1/2
	$R = O(h^2)$	a b c	0 1/2 1/2	1/2 0 1/2	1/2 1/2 0	1/6 1/6 1/6
	$R = O(h^4)$	a b c d e f g	1/3 1 0 0 0 1/2 1/2	1/3 0 1 0 1/2 0 1/2	1/3 1 0 1 1/2 1/2 0	27/120 3/120 3/120 3/120 8/120 8/120 8/120

Результати досліджень

На рис. 1. показані графіки переміщень в перерізах для різних переміщень у отримані на сітці, яка мітить 2352 квадратичні елементи. Результати розрахунку напружень для цих випадків зображено на рис. 2.

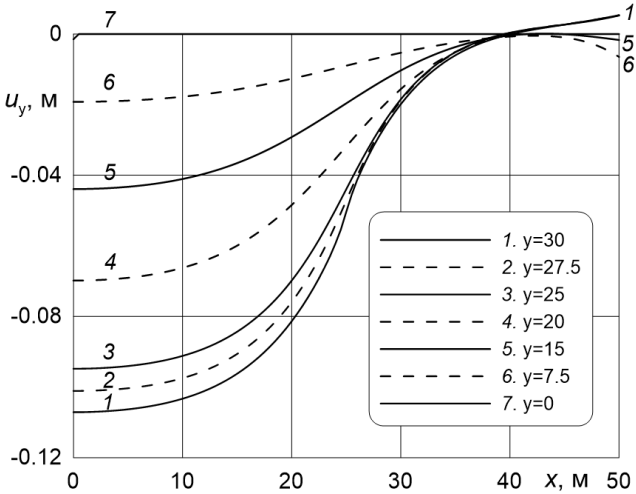


Рис. 1. Графіки переміщень на сітці

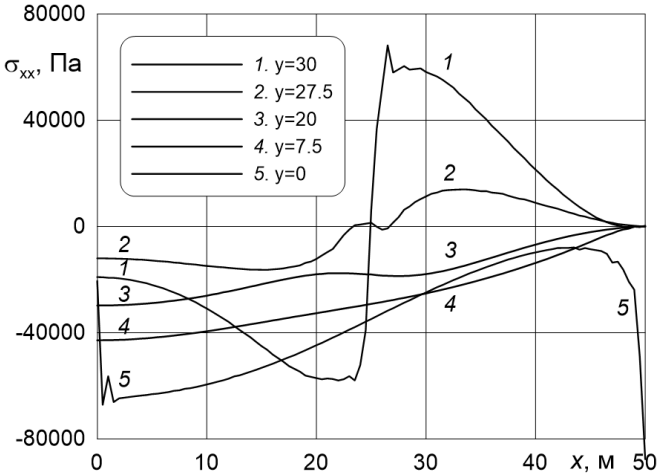


Рис. 2. Розрахунок напружень

В якості прикладу використання різних трикутних скінченних елементів було проведено моделювання напружень та деформацій основи в районі будівництва з використанням методу скінченних елементів. На основі отриманих результатів можна зробити наступні висновки.

Для початку, підтвердженням вірності обрахунків слугує те, що напруження σ_{yy} на поверхні навантаження дорівнюють зовнішньому навантаженню.

По-друге, згущення сітки як у випадку лінійних вузлів так і у випадку квадратичних позитивно впливає на одержані результати. Збіжність згідно ущільнення сітки елементів проілюстрована на графіках для переміщень та напружень отриманих на вказаних сітках елементів.

По-третє, я зробила висновок, що результати одержані за допомогою квадратичних вузлів є точнішими, ніж з тими самими параметрами, але лінійними вузлами. Адже вже при 76 квадратичних елементах результат майже збігається з тим, що одержаний для 2352 лінійних елементів. Це є достатньо суттєвим про обрахунку навантажень.

Список літератури

1. Скворцов А.В. Триангуляция Делоне и её применение. – Томск: Изд-во Том. ун-та, 2002. – 128 с.
2. В. С. Андреев Математическое моделирование внецентренно нагруженных фундаментов на слоистых основаниях методом конечных элементов./ В. С. Андреев, А. Л. Тютькин, Т. А. Селихова // Наука та прогресс транспорту. – 2004. – №4. – С.183 – 187.
3. Triangle: A Two-Dimensional Quality Mesh Generator and Delaunay Triangulator (cmu.edu) [Електронний ресурс]. – Режим доступу: http://www.cs.cmu.edu/~quake/triangle.html (програма Triangle)
4. Дутчин М. Дослідження точності визначення осідань фундаментів інженерних споруд на стадії ущільнення ґрунтів / Дутчин М., Грицюк Т., Ничвид М. // Збірник наукових праць Західного геодезичного товариства УТГК. – 2014. – Випуск I (27). – С. 45 – 56.

АВТОРИЗАЦІЯ КОРИСТУВАЧА ЗА ГОЛОСОМ

Дана Зарецька, Микола Баранов, Сергій Іванов
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
danazaretska@gmail.com

Вступ

Вирішення задач ідентифікації та аутентифікації людини за голосом спираються на створення методів, що базуються на знаходженні відмінностей між мовцями. Системи розпізнавання мовців використовують голос людини як ключовий інструмент для перевірки особистості. Разом з методами аудіоаналізу, часто використовують моделі на основі штучних нейронних мереж.

Тема дослідження: обробка та розпізнавання аудіо-сигналів. Створення алгоритму авторизації людського голосу.

Мета дослідження: розробити модель аудіо-аутентифікації голосу людини на базі штучних нейронних мереж.

Методи розпізнавання мовців

Задачі розпізнавання мовця поділяються на дві групи: *контекстно-залежні* (text-dependent methods) та *контекстно-незалежні* (text-independent methods)[1].

Запропонований алгоритм

Запропонований алгоритм бере за основу *використання зображень мел-спектрограм* як голосових ознак та *нейронну модель-класифікатор* для розпізнавання.

Набір даних, що використовувався

Для дослідження використано набір даних Speaker Recognition Audio Dataset fff61aed-e, представлений на платформі для змагань з аналітики та передбачувального моделювання Kaggle у 2018. Дані складаються з 50-ти аудіодоріжок, кожна з яких містить запис голосу однієї людини тривалістю 60 хвилин. Текст, що промовляється, є унікальним для кожного мовця, тому для таких даних задача полягатиме у *контексно-незалежному розпізнаванні*.

Обробка даних

Над кожним аудіозаписом відбувся процес видалення тиші. Вирізалися ті часові проміжки, в яких максимальна гучність не перевищувала 15 дБ. Отримані дані було поділено на рамки тривалістю 4 секунди кожна. Для

тренування класифікаційної моделі використовувалися зображення мел-спектрограм[2].

Звичайна *спектрограма* є відображенням спектру частот у часі, демонструє залежність між силою спектруму та часом[3]. *Мел-спектрограма* утворена за допомогою певного логарифмування значень по осі Y . Зазвичай перетворення лінійної шкали у мел-шкалу відбувається за такою формулою: $H(f) = 1127 \cdot \ln(1 + (f / 700))$.

Запропонована архітектура нейронної мережі

В ролі класифікаційної моделі, використано сіамську нейронну мережу[4][5], яка приймає на вхід позитивні та негативні пари. Позитивними парами у випадку даної задачі будуть дві мел-спектрограми, згенеровані з голосів однієї людини, а негативними - ті, що згенеровані з голосів двох різних мовців.

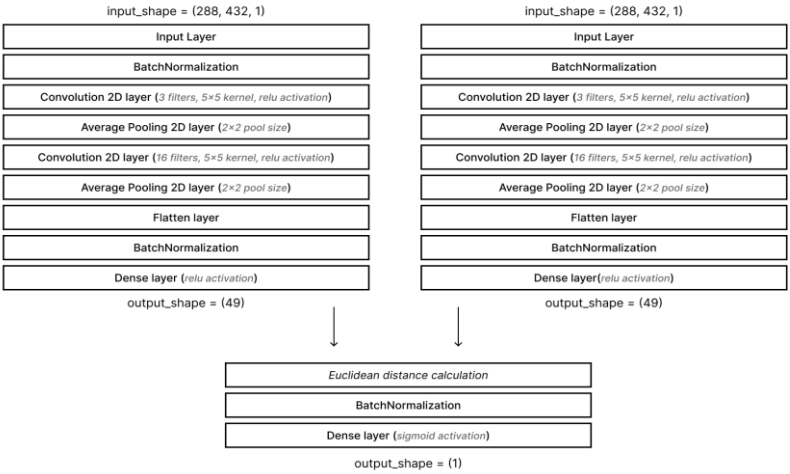


Рис. 1. Архітектура використаної сіамської нейронної мережі

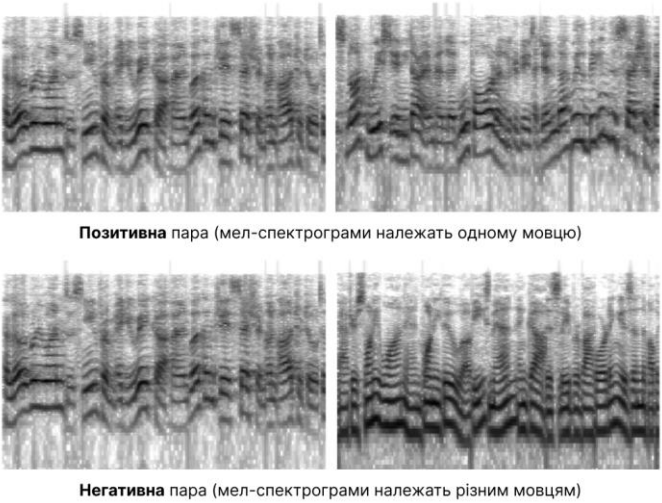


Рис. 2. Приклад позитивної та негативної пари, що були використані для тренування мережі

В ролі функції-оптимізатора для моделі обраний Adam. Функцією втрат є Triplet Loss.

Отримані результати

Табл. 1. Результати

Метрика	Accuracy	Recall	Precision	F1	MCC
Значення	0.9792	1.0	0.96	0.9796	0.9592

Нижче представлено графіки того, як змінюється *loss*-функція та *Accuracy* на кожній епосі. Суцільною лінією позначено таку зміну на тренувальних даних, пунктиром - на валідаційних:

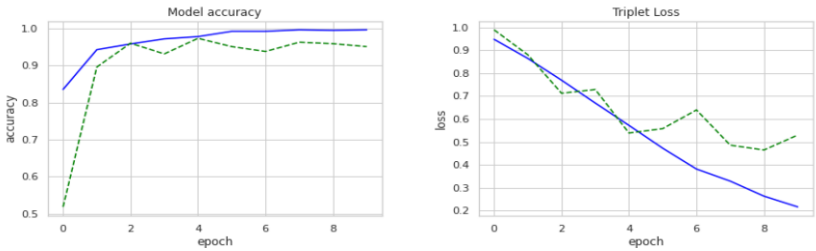


Рис. 3. Графіки зміни *loss*-функції та *Accuracy* в процесі тренування мережі

Наведений алгоритм аудіо-аутентифікації демонструє хороші результати, проте має певні обмеження. Присутність шумів в аудіодоріжці спотворює процес видалення тиші, рамки можуть бути представлені некоректно, що ускладнить процес правильної авторизації. Способом вирішення цієї проблеми може бути застосування певних фільтрів до етапу видалення тиші. Другою особливістю є те, що присутність кількох голосів в аудіозаписі непередбачувано впливає на роботу класифікатора.

Для програмної реалізації використовувалася мова Python, бібліотеки librosa для роботи з аудіосигналом, numpy для опрацювання векторних даних, tensorflow та keras для роботи з нейронними мережами.

Список літератури

1. Mobiny, M. Najarian Text-Independent Speaker Verification Using Long ShortTerm Memory Networks / Department of Electrical and Computer Engineering, University of Houston, September 2018
2. Lam Pham, Ian McLoughlin, Huy Phan, Ramaswamy Palaniappan A Multispectrogram Deep Neural Network for Acoustic Scene Classification / Detection and Classification of Acoustic Scenes and Events, 2019
3. Heckbert P.Fourier Transforms and the Fast Fourier Transform (FFT) Algorithm Feb. 1995
4. G. Koch, R. Zemel, R. Salakhutdinov Siamese Neural Networks for One-shot Image Recognition / Department of Computer Science, University of Toronto [https://www.cs.cmu.edu/ rsalakhu/papers/oneshot1.pdf]
5. Tanish Baranwal Facial Re-Identification Using Siamese Nets, 2020 [https://towardsdatascience.com/facial-re-identification-using-siamese-netsd36df39da7c0]

ЧИСЛОВЕ МОДЕЛЮВАННЯ МЕТОДОМ СКІНЧЕННИХ ЕЛЕМНТІВ ЗАДАЧ ТЕПЛОПРОВІДНОСТІ

Катерина Старовойт

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

Kateryna.Starovoit@lnu.edu.ua

Вступ

Метод скінченних елементів (МСЕ) є одним з часто використовуваних чисельних методів. Широке застосування та актуальність цього методу забезпечуються успішним розв'язанням прикладних задач довільної складності запропонованим алгоритмом. У даному дослідженні буде розглянуто застосування МСЕ для розв'язку задач теплопровідності.

1. Постановка крайової та варіаційної задачі

Розглянемо задачу з рівнянням дифузії-адвекції-реакції вигляду: знайти функцію u = u(x), яка є розв'язком рівняння

$$-\frac{d}{dx}\left[\mu(x)\frac{du(x)}{dx}\right]+\beta(x)\frac{du(x)}{dx}+\sigma(x)u(x)=f(x)\ \forall x\in\Omega=(0,1) \qquad (1.1)$$

з такими крайовими умовами: u(0) = 0, u(1) = 0.

Тут $\mu = \mu$ (x), $\beta = \beta$(x), $\sigma = \sigma(x)$ та f= f (x) – задані функції такі, що $\mu(x) > 0$, $\sigma(x) \geq 0$ майже скрізь в Ω і μ, β, $\sigma \in L^{\infty}(\Omega)$, $f \in L^2(\Omega)$.

Варіаційне формулювання задачі (1.1) має вигляд:

Знайти $u \in V$ таку, що $a(u,v) := < l, v >\ \forall v \in V$, де

$$\begin{cases} V := H_0^1(\Omega) = \{v \in H^1(\Omega): v(0)=0, v(1)=0\}, \\ c(u,v) := \int_0^1(-\mu u'v' + \beta u'v + \sigma u)v dx \quad \forall u,v \in V, \\ \qquad < l, v > := \int_0^1 f v dx \qquad \forall v \in V. \end{cases} \qquad (1.2)$$

1.1 Скінченно елементна дискретизація. Вузлами $0 = x_0 < x_1 < \cdots < x_{N-1} < x_N = 1$ поділимо відрізок на скінченні елементи $K_i = (x_i, x_{i+1})$, на яких визначимо кусково-лінійні функції Куранта[3]: $\{\varphi_i(x)\}_{i=1}^{N-1}$

Апроксимацію МСЕ запишемо у вигляді лінійної комбінації:

$$u_h(x) = \sum_{i=1}^{N-1} u_{i+1/2}(x) = \sum_{i=1}^{N-1} q_i\varphi_i(x) \qquad (1.3)$$

з вектором невідомих коефіцієнтів $q = \{q_i\}_{i=1}^{N-1}$.

Коефіцієнти $\mathbf{q_1}, \dots\ \mathbf{q_{N-1}}$ розвинення $\mathbf{u_h} = \mathbf{u_h}(\boldsymbol{x})$обчислюються як розв'язки системи лінійних алгебричних рівнянь МСЕ, див.[3].

1.2 Апостеріорний оцінювач похибки і адаптивна схема МСЕ. Щоб наближено обчислити похибку знайденої апроксимації(1.3), використано кусково-квадратичний апостеріорний оцінювач похибки МСЕ вигляду[3]:

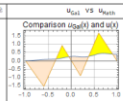
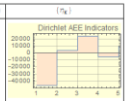
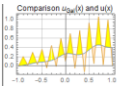
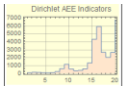
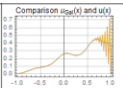
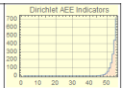
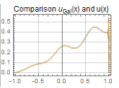
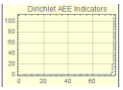
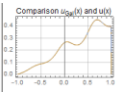
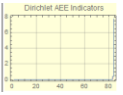
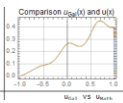
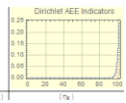
$$\mathbf{u(x) - u_h(x) \approx \varepsilon_h(x) := \sum_{i=1}^{N-1} \lambda_{i+1/2}\, b_{i+1/2}(x)}, \qquad (1.4)$$

де $\mathbf{b_{i+1/2}(x) := 4\left[1 - \omega_{i+1/2}(x)\right]\omega_{i+1/2}(x)\ \forall x \in x[x_i, x_{i+1}], i = 1, \dots, N-1[}$

На основі цього АОП побудовано h-адаптивну схему скінченних елементів.

1.3 Результати числових експериментів. Програмна реалізація розв'язку задачі (1.1) з обчисленими основними апріорними та апостеріорними оцінками похибки показала неефективінсть рівномірної схеми для сингулярно-збурених задач. Натомість використання адаптивної схеми дозволяє розв'язати задачу з точністю 10% та використанням 101 скінченного елемента.

Таблиця 1. Збіжність послідовності h-адаптивного МСЕ

step	n	$\|u_{h1}\|_\infty$	$\|u_{h1}\|_2$	$\|e_{h1}\|_\infty$	$\|e_{h1}\|_2$	$\frac{\sigma_h^2 = \|e_{h1}\|_\infty / \|u_{h1}\|_\infty 100\ \%}{}$	$\frac{\sigma_h^2 = \|e_{h1}\|_2 / \|u_{h1}\|_2 100\ \%}{}$	u_{h1} vs u_{exa}	Convergence rate $\|e_{h1}\|_2$	rate $\|e_{h1}\|_2$	$\{\eta_k\}$
1	5	1.15239	7.38053	1.04839	10.5892	90.9751	143.475		1.20638	0.636271	
3	20	0.569665	9.94018	0.317008	12.3924	55.6482	124.67		-0.193322	1.24688	
5	55	0.375506	8.04083	0.0564794	10.2101	15.0409	126.978		0.139878	1.64699	
7	77	0.371216	7.97808	0.010963	7.40093	2.95327	92.7658		1.72003	5.48735	
9	84	0.370951	7.96917	0.00108091	2.74355	0.291388	34.4271		24.1637	28.722	
11	101	0.371266	7.97358	0.000348415	0.716312	0.093845	8.98357		4.58254	4.60472	
step	n	$\|u_{h1}\|_\infty$	$\|u_{h1}\|_2$	$\|e_{h1}\|_\infty$	$\|e_{h1}\|_2$	$\frac{\sigma_h^2 = \|e_{h1}\|_\infty / \|u_{h1}\|_\infty 100\ \%}{}$	$\frac{\sigma_h^2 = \|e_{h1}\|_2 / \|u_{h1}\|_2 100\ \%}{}$	u_{h1} vs u_{exa}	Convergence rate $\|e_{h1}\|_2$	rate $\|e_{h1}\|_2$	$\{\eta_k\}$

Натомість апроксимації МСЕ на рівномірно згущуваних вдвічі сітках показало, що досягнення рівня tol = 10% можливе при наявності N = 1024.

2. Початково-крайова та варіаційна задачі про теплову хвилю

2.1 Формулювання задачі . Тут розглянуто задачу поширення теплової хвилі в такій редакції[1]: знайти температуру $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{r}, \boldsymbol{t})$ таку, що:

$$\begin{cases} \boldsymbol{\tau_0}\frac{\partial^2 \mathrm{T}}{\partial \mathrm{t}^2} + \frac{\partial \mathrm{T}}{\partial \mathrm{t}} = \frac{\mathrm{k}}{\mathrm{r}}\frac{\partial}{\partial \mathrm{r}}\left(\mathrm{r}\frac{\partial \mathrm{u}}{\partial \mathrm{r}}\right) + \mathbf{f} \text{ в } \boldsymbol{\Omega} \times (\mathbf{0}, \mathbf{T}], \\ \mathbf{u}(\mathbf{0}, \mathbf{t}) = \mathbf{p}(\mathbf{t}),\ \mathbf{u}(\mathbf{1}, \mathbf{t}) = \mathbf{q}(\mathbf{t}) \quad \forall \mathbf{t} \in [\mathbf{0}, \mathbf{T}], \\ \mathbf{u}(\mathbf{r}, \mathbf{0}) = \mathbf{u_0}(\mathbf{r}), \quad \frac{\partial \mathbf{u}}{\partial \mathbf{t}}(\mathbf{r}, \mathbf{0}) = \mathbf{v_0}(\mathbf{r}) \quad \forall \mathbf{r} \in \boldsymbol{\Omega} = (\mathbf{0}, \mathbf{1}). \end{cases} \tag{2.1}$$

Тут $\boldsymbol{\tau_0} \geq \mathbf{0}$ – час релаксації, $\boldsymbol{k} > \mathbf{0}$ – коефіцієнт температуропровідності. Відповідне до (2.1) варіаційне формулювання задачі має вигляд: знайти $\boldsymbol{u} \in \boldsymbol{L^2}\left(\mathbf{0}, \boldsymbol{T}; \boldsymbol{H_0^1}(\boldsymbol{\Omega})\right)$ таку, що:

$$\begin{cases} \boldsymbol{\tau_0}(\boldsymbol{u}''(\mathbf{t}) + \mathbf{u}'(\mathbf{t}), \mathbf{v}) + \mathbf{c}(\mathbf{u}(\mathbf{t}), \mathbf{v}) = (\mathbf{f}(\mathbf{t}), \mathbf{v}) \ \ \forall \mathbf{t} \in (\mathbf{0}, \mathbf{T}], \\ (\mathbf{T}(\mathbf{0}, \mathbf{t}) - \mathbf{p}(\mathbf{t}), \mathbf{v}) = \mathbf{0}, (\mathbf{T}(\mathbf{1}, \mathbf{t}) - \mathbf{q}(\mathbf{t}), \mathbf{v}) = \mathbf{0}, \ \forall \mathbf{v} \in \boldsymbol{H_0^1}(\boldsymbol{\Omega}), \end{cases} \tag{2.2}$$

де $\boldsymbol{u}'(\boldsymbol{t}) = \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{t}}(\boldsymbol{r}, \boldsymbol{t}), \qquad \boldsymbol{u}(\boldsymbol{t}) \equiv \boldsymbol{u}(\boldsymbol{r}, \boldsymbol{t}),$

$(\boldsymbol{u}, \boldsymbol{v}) := \int_0^1 \boldsymbol{uvrdr},\ \boldsymbol{c}(\boldsymbol{u}, \boldsymbol{v}) := \int_0^1 \boldsymbol{k}\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{r}}\frac{\partial \boldsymbol{v}}{\partial \boldsymbol{r}}\boldsymbol{rdr}.$

2.2 Дискретизація задачі. Вживаючи кусково-лінійний базис $\{\boldsymbol{\varphi_i}(\boldsymbol{r})\}_{\boldsymbol{i=1}}^{\boldsymbol{N-1}}$, визначимо напівдискретну апроксимацію Гальоркіна у вигляді

$$\boldsymbol{u_h}(\boldsymbol{r}, \boldsymbol{t}) = \sum_{\boldsymbol{i=1}}^{\boldsymbol{N-1}} \boldsymbol{\varphi_i}(\boldsymbol{r})\boldsymbol{q_i}(\boldsymbol{t}).$$

Щоб знайти наближення до вектора $\boldsymbol{q}(\boldsymbol{t}) := \{\boldsymbol{q_i}(\boldsymbol{t})\}_{\boldsymbol{i=1}}^{\boldsymbol{N-1}}$, покриємо часовий проміжок [0;T] сіткою вузлів $\boldsymbol{t_j} = \boldsymbol{j}\Delta\boldsymbol{t} = \boldsymbol{j}\frac{\boldsymbol{T}}{\boldsymbol{N}(\boldsymbol{T})}, \boldsymbol{j} = \mathbf{0}, \dots, \boldsymbol{N}(\boldsymbol{T})$, і на кожному кроці інтегрування будемо шукати наближення до $\boldsymbol{q}(\boldsymbol{t})$ у вигляді

$$\boldsymbol{q}(\boldsymbol{t}) \approx \boldsymbol{q_j}(\boldsymbol{t}) := \boldsymbol{q}^{\boldsymbol{j}} + \Delta\boldsymbol{t}\boldsymbol{\omega}(\boldsymbol{t})\dot{\boldsymbol{q}}^{\boldsymbol{j}} + \frac{\mathbf{1}}{\mathbf{2}}[\Delta\boldsymbol{t}\boldsymbol{\omega}(\boldsymbol{t})]^{\mathbf{2}}\ddot{\boldsymbol{q}}^{\boldsymbol{j+1/2}},$$

де $\boldsymbol{q}^{\boldsymbol{j}}, \dot{\boldsymbol{q}}^{\boldsymbol{j}}, \ddot{\boldsymbol{q}}^{\boldsymbol{j+1/2}} \in \mathbb{R}^{\boldsymbol{N-1}}$ – невідомі вектори. Для їх визначення скористаємося однокроковою рекурентною схемою:

задано $\dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}} \in \mathbb{R}^{\boldsymbol{N-1}}, \Delta\boldsymbol{t} > \mathbf{0}, \boldsymbol{\gamma}, \boldsymbol{\beta} \in [\mathbf{0}; \mathbf{1}]$; знайти вектор $\ddot{\boldsymbol{q}}^{\boldsymbol{j+1/2}} \in \mathbb{R}^{\boldsymbol{N-1}}$ такий, що

$$\left[(\boldsymbol{\tau_0} + \Delta\boldsymbol{t}\boldsymbol{\gamma})\boldsymbol{G} + \frac{\mathbf{1}}{\mathbf{2}}\Delta\boldsymbol{t}^{\mathbf{2}}\boldsymbol{\beta}\boldsymbol{C}\right]\ddot{\boldsymbol{q}}^{\boldsymbol{j+1/2}} = \boldsymbol{F}^{\boldsymbol{j+1/2}} - \boldsymbol{G}\dot{\boldsymbol{q}}^{\boldsymbol{j}} - \boldsymbol{C}[\boldsymbol{q}^{\boldsymbol{j}} + \Delta\boldsymbol{t}\boldsymbol{\gamma}\dot{\boldsymbol{q}}^{\boldsymbol{j}}] \tag{2.3}$$

та $\dot{\boldsymbol{q}}^{\boldsymbol{j+1}} = \dot{\boldsymbol{q}}^{\boldsymbol{j}} + \Delta\boldsymbol{t}\ddot{\boldsymbol{q}}^{\boldsymbol{j+1/2}}, \boldsymbol{q}^{\boldsymbol{j+1}} = \dot{\boldsymbol{q}}^{\boldsymbol{j}} + \Delta\boldsymbol{t}\dot{\boldsymbol{q}}^{\boldsymbol{j}} + \frac{\mathbf{1}}{\mathbf{2}}\Delta\boldsymbol{t}^{\mathbf{2}}\ddot{\boldsymbol{q}}^{\boldsymbol{j+1/2}}, \boldsymbol{j} = \mathbf{0}, \mathbf{1} \dots \boldsymbol{N}(\boldsymbol{T}).$

Тут $\boldsymbol{G} = \{\boldsymbol{\varphi_i}, \boldsymbol{\varphi_j}\}_{\boldsymbol{i,j=1}}^{\boldsymbol{N-1}}, \boldsymbol{C} = \{\boldsymbol{c}(\boldsymbol{\varphi_i}, \boldsymbol{\varphi_j})\}_{\boldsymbol{i,j=1}}^{\boldsymbol{N-1}}, \boldsymbol{F}^{\boldsymbol{j+1/2}} = \{(\mathbf{f}\left(\boldsymbol{t_j} + \frac{\mathbf{1}}{\mathbf{2}}\Delta\boldsymbol{t}\right), \boldsymbol{\varphi_i})\}_{\boldsymbol{i,j=1}}^{\boldsymbol{N-1}}.$

За вибору $\frac{\mathbf{1}}{\mathbf{2}} \leq \boldsymbol{\gamma} \leq \boldsymbol{\beta}$ ОРС безумовно стійка.

2.3 Результати числових експериментів. Програмну реалізацію схеми(2.3) здійснено мовою Python з використанням бібліотеки fenics[2]. Нижче наведено результати експерименту для задачі (2.1) з даними, взятими зі статті[1]. Одержані апроксимації добре узгоджуються з числовими значеннями, наведеними в згаданій статті.

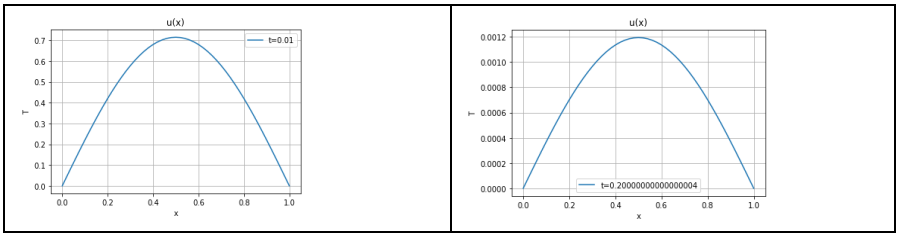


Рис. 1.Обчислення виконано з $\Delta t = 0.001, N = 40$.

Висновки

В порівнянні з апроксимаціями на рівномірних сітках h-адаптивні наближення дозволяють розв'язати задачу(1.1) з допустимою похибкою в 10% з використанням локально згущеної сітки зі 101 скінченних елементів, демонструючи порядки швидкості збіжності $\boldsymbol{rate}(||\boldsymbol{e_h}||) \geq \mathbf{4}$.

Стосовно однокрокової рекурентної схеми (2.3), то використання МСЕ дозволяє одержати наближені розв'язки, які добре узгоджуються з результатами статті[1], одержаними із застосуванням методу скінченних різниць.

Використані джерела

1. Madhukar A. Heat conduction in porcine muscle and blood: experiments and time-fractional telegraph equation model / A. Madhukar, Y. Park, W. Kim, H. J. Sunaryanto, R. Berlin, L. P. Chamorro, J. Bentsman, M. Ostoja-Starzewski // J. R. Soc. Interface 16:20190726. http://dx.doi.org/10.1098/rsif.2019.0726
2. Logg A. Automated Solution of Differential Equations by the Finite Element Method The FEniCS Book / A. Logg, K.-A. Mardal, G. N. Well. Springer, 2012.
3. Абрамов Є. Кусково-лінійні апроксимації h-адаптивного методу скінченних елементів для одновимірних крайових задач / Абрамов Є., Ліпіна О, Шинкаренко Г., Ямелинець А. // Вісн. Львів. ун-ту. Сер. прикл. матем. та інформ. 2006. Вип. 11. С. 3–18

ВИКОРИСТАННЯ ГЕНЕТИЧНИХ АЛГОРИТМІВ ДЛЯ ЧИСЕЛЬНОГО РОЗВ'ЯЗУВАННЯ ДВОВИМІРНИХ ІНТЕГРАЛЬНИХ РІВНЯНЬ ПЕРШОГО РОДУ

Роман Куновський

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

romankunovskyi@gmail.com

В даній роботі за допомогою генетичних алгоритмів буде наведено алгоритм для чисельного розв'язування двовимірних інтегральних рівнянь першого роду.

Нехай задана границя $\Gamma \in C^2$, як зображено на (рис. 1), для якої відомо параметричне подання.

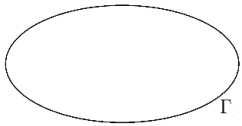


Рис. 1. Вигляд границі

Розглянемо двовимірне інтегральне рівняння першого роду, яке має наступний вигляд:

$$\int_{\Gamma} \varphi(y) H(x,y) ds(y) = f(x), \;\; x \in \Gamma \tag{1}$$

Задача полягає в тому, щоб знайти функцію $\varphi(y)$, коли ядро $H(x,y)$ і функція $f(x)$ є відомими неперервними функціями. Існування та єдиність розв'язку рівняння (1) базується на теорії Рісса-Шаудера, яка описана в [1]. Відомо з [5], що розвязок задачі є нестійким, тому дана задача є некоректною.

Відоме параметричне задання границі:

$$\Gamma := \{x(t) = (x_1(t), x_2(t)), \; t \in [0,2\pi]\}, \tag{2}$$

де $x: \mathbb{R} \longrightarrow \mathbb{R}^2$, 2π-періодична функція з $|x'(t)| > 0 \;\; \forall t \in [0,2\pi]$.

Параметризований вигляд інтегрального рівняння (1):

$$\int_0^{2\pi} \psi(\tau) K(t,\tau) d\tau = f\big(x(t)\big), \; t \in [0,2\pi] \tag{3}$$

де $\psi(\tau) = \varphi(x(\tau))$, а $K(t,\tau) = H\big(x(t), x(\tau)\big)|x'(\tau)|$, де $t, \tau \in [0,2\pi]$.

Отже, за допомогою параметризації було перейдено з двовимірного до одновимірного інтегрального рівняння, що беззаперечно є зручніше при чисельному розв'язуванні.

Інтегральне рівняння (3) будемо розвязувати за допомогою генетичних алгоритмів. Генетичний алгоритм – це в першу чергу еволюційний алгоритм. Розглянемо детальніше основні компоненти генетичних алгоритмів, які наведені в [2].

Створення нової популяції. Зазвичай початкова популяція містить згенеровані випадковим чином індивіди.

Селекція – це вибір індивідів на основі значень функції оцінки.

Схрещення – це комбінування хромосом батьків індивідів.

Мутація – це зміна хромосом з заздалегідь визначеними операціями.

Формування нової популяції – це вибір частки тих, хто «піде далі».

Ці кроки повторюються до тих пір поки не виконуються критерії зупинки. Збіжність генетичних алгоритмів описана в [2].

Ми визначаємо наближення до розв'язку інтегрального рівняння (5) наступним чином:

$$\psi_n(t) = \sum_{i=1}^{n} a_i \mu_i(t), \tag{4}$$

де a_i - невідомі коефіцієнти, μ_i – відомі базисні функції.

Підставивши (4) в (3), отримаємо $r_n(x)$ - залишковий елемент.

$$r_n(t) = \int_0^{2\pi} K(t,\tau)\psi(\tau)d\tau - f\big(x(t)\big) \approx \int_0^{2\pi} K(t,\tau)\sum_{i=1}^{n} a_i\mu_i(t)d\tau - f\big(x(t)\big) =$$

$$= \sum_{i=1}^{n} a_i \int_0^{2\pi} K(t,\tau)\mu_i(\tau)d\tau - f\big(x(t)\big) = \sum_{i=1}^{n} a_i b_i(t) - f\big(x(t)\big),\ t \in [0,2\pi].$$

Зазначимо, що $b_i(t) \approx \int_0^{2\pi} K(t,\tau)\mu_i(\tau)d\tau$. Для чисельного розв’язування даного інтегралу можна застосувати складену квадратуру трапецій.

Наша мета – знайти $a_i,\ i = 1,2,\dots,n$ такі, що $r_n(t) = 0$. Для отримання такого результату нам необхідно звести задачу до пошуку мінімуму $|r_n(t)|$. Двовимірне інтегральне рівняння першого роду є некоректно поставленою задачею. Такі задачі розв’язуються методом регуляризації. В нашому випадку пропонується застосувати мінімізуючий функціонал Тіхонова і звести нашу задачу до наступної:

$$Minimize\ \ |\tilde{r}_n(t)| = \left(\sum_{i=1}^{n} a_i b_i(t) - f(x(t))\right)^2 + \alpha \sum_{i=1}^{n} a_i^2,\ t \in [0,2\pi), \quad (5)$$

де α – параметр регуляризації, який зазвичай дорівнює малому числу.

Перейдемо до опису алгоритму. Зазначимо, що формула (5) описуватиме функцію оцінки індивідів.

Крок 1. Створюється початкова популяція індивідів, кожен з яких зберігає n коефіцієнтів a_i, представлених бітовими стрічками.

Крок 2. Якщо оцінюється початкова популяція, то її найкращий індивід записується як альфа індивід, інакше він порівнюється з існуючим альфа індивідом і кращий з них запам'ятовуємо.

Крок 3. Перевіряємо умови виходу, які запропоновані нижче. Якщо відповідні умови здійснюються, то альфа індивід вважається розв'язком і алгоритм закінчує роботу.

Крок 4. За допомогою турнірного методу вибираємо індивіди, з яких буде формуватися нова популяція.

Крок 5. Індивіди переможці мають шанс схрещуватися або мутувати. Якщо індивід був обраний для схрещування або мутації, тоді кожна його хромосома має шанс схреститися з відповідною хромосомою партнера або мутувати. Після алгоритм переходить на крок 2.

Розглянемо три варіанти умов виходу, які можна використовувати незалежно або комбінувати:

1. Знайдений альфа індивід, для якого оцінка менша за E, де E – це мале дійсне число, обране на початку алгоритму.
2. Знайдений альфа індивід не покращується протягом багатьох популяцій. Це може означати, що був знайдений найкращий можливий індивід.
3. Кількість ітерацій досягла заздалегідь обраного максимуму.

Наведемо результати обчислень. Розглянемо границю Γ_1, яку зображено на (рис. 2), і наступне інтегральне рівняння з відповідним точним розв'язком:

$$\int_{\Gamma_1} \varphi(x)(x_1x_2 + y_1y_2)^2 ds(y) = \pi x_1^2 x_2^2,\ x \in \Gamma_1$$

$$\varphi(y) = 0.5 + (y_1 - y_2)^2 - 4y_1^2y_2^2.$$

За допомогою алгоритму ми знайшли $\varphi_*(y)\ \approx\ \varphi(y)$:

$\varphi_*(y) = 0.498962 + 0.995855((y_1 - y_2)^2 - 4y_1^2y_2^2)\ \cdot$

Як ми бачимо з результатів чисельних експериментів, генетичні алгоритми дали досить точний розв'язок двовимірного інтегрального рівняння

першого роду. Також видно, що алгоритм доволі швидко знаходить хороший результат, який протягом наступних ітерацій зазнає незначних покращень.

В даній роботі було продемонстровано ефективність генетичних алгоритмів для знаходження чисельного розв'язку двовимірних інтегральних рівнянь першого роду.

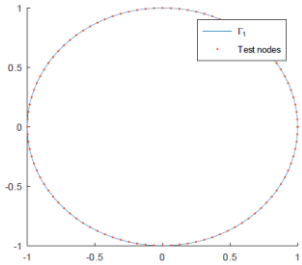


Рис. 2. Вигляд границі Γ_1

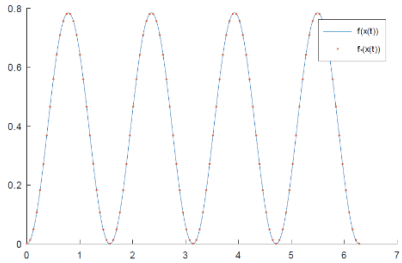


Рис. 3. Точне значення функції і значення функції при підстановці $\varphi_*(y)$

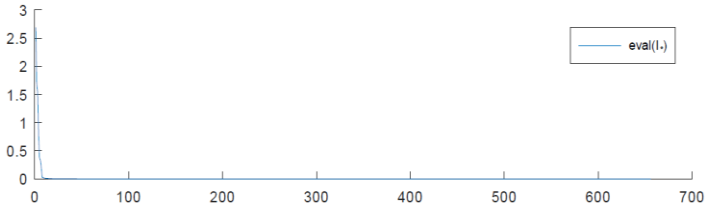


Рис. 4. Зміна значення функції оцінки найкращого індивіда в кожній з популяцій

Список літератури

1. *Rainer Kress.* Linear Integral Equations (Third edition) // Springer.– 2013.– С. 297-321.
2. *Zbigniew Michalevicz.* Genetic algorithm + data structures = evolutions programs // Springer.– 1996.– С. 11-88.
3. *Roman Chapko*. On the use of an integral equation approach for the numerical solution of a Cauchy problem for Laplace equation in a doubly connected planar domain / Roman Chapko, B. Tomas Johansson, Yuriy Savka // Applied Mathematics in Science and Engineering (AMSE) .– 2014.
4. *Roman Chapko.* An alternating boundary integral based method for inverse potential flow around immersed bodies / Roman Chapko, B. Tomas Johansson // Inverse Problems and Imagine.– 2008.– С. 317-333.
5. *Юрій Чорноіван.* Конспект лекцій з курсу інтегральних рівнянь та елементів функціонального аналізу / 2017р.– С. 147-159.

ЧИСЕЛЬНЕ РОЗВ'ЯЗУВАННЯ ОБЕРНЕНОЇ ЗАДАЧІ ТЕПЛОПРОВІДНОСТІ ПАРАЛЕЛЬНИМИ ГЕНЕТИЧНИМИ АЛГОРИТМАМИ

Іван Винявський

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

Ivan.Vyniavskyi@lnu.edu.ua

Вступ

Обернена задача теплопровідності – це задача визначення зовнішнього коефіцієнту тепловіддачі для даних, отриманих шляхом спостережень за історією температури в одній або більше внутрішніх точок. Вона протиставляється прямій задачі теплопровідності, яка полягає у визначенні внутрішнього температурного поля для заданого набору граничних умов.

Крайові умови повинні бути відомими, у нашому випадку вони описуються зовнішньою температурою та коефіцієнтом тепловіддачі h на межі тіла. Розрахунок температурних полів – класична проблема, для обрахунку можна використати метод скінченних різниць або метод скінченних елементів.

Задача стає більш складною, коли внутрішня температура є відомою, а відповідні граничні умови – невідомі, це і є обернена задача, яка є некоректна з математичної і чисельної точки зору. В даній роботі таку задачу ми будемо розв'язувати за домогою генетичних алгоритмів.

Постановка задачі

Обернена задача теплопровідності:

$$T_t(x,t) = T_{XX}(x,t),\ 0 < x < 1,\ 0 < t < t_M$$
$$T(x,0) = f(x),\ 0 \leqslant x \leqslant 1$$
$$T(0,t) = p(t),\ 0 \leqslant t \leqslant t_M$$
$$T(1,t) = q(t),\ 0 \leqslant t \leqslant t_M$$

умова перевизначеності

$$T(a,t) = s(t),\ 0 \leqslant t \leqslant t_M, 0 < a < 1$$

де $f(x)$ відома неперервна функція, $p(t)$ і $q(t)$ є нескінечно диференційовані функції і t_M являє собою остаточний час існування для еволюційної проблеми в часі, при тому функція $q(t)$, невідома. Функцію $q(t)$

можна визначити за допомогою вимірювань внутрішньої температури. Задача може бути вирішена за допомогою методу найменших квадратів. Тоді функція витрат може визначатися як сума квадратичних різниць між вимірюваними температурами та розрахунковими значеннями T шляхом врахування наближень про розрахункові значення $q(t)$.

Різницева апроксимація прямої задачі теплопровідності

Під час застосування генетичних алгоритмів до оберненої задачі потрібно розв'язати пряму задачу теплопровідності, тому для її чисельного розвязання ми використовуємо метод Кранка-Ніколсона. За допомогою даного методу ми отримуємо різнецеву апроксимацію. Дану апроксимацію ми записуємо у вигляді СЛАР та розвязуємо за допомогою методу Гауса.

Парелельні генетичні алгоритми

За допомогою паралельних генетичних алгоритмів часто можна отримати кращі показники, ніж за допомогою генетичних алгоритмів, які використовують одну велику паніктичну сукупність. Всі відомі паралельні генетичні алгоритми можливо розділити на два основних класи: однопопуляційні та багатопопуляційні. Однопопуляційні ПГА реалізують паралельне обчислення тільки функції придатності, а генетичні операції виконуються централізовано головним процесом.

В багатопопуляційних ПГА здійснюється формування множини незалежних популяцій (островів) та реалізується пошук рішень окремими ГА в кожній з популяції в межах окремих процесів. Кожний з островів може бути ізольованим, або зв'язаним з іншими островами за допомогою оператора міграції. Така модель більш дотримується законів природної еволюції.

Схема генетичного алогритму

Етапи генетичного алгоритму для визначення $q(t)$ можна розділити на наступні кроки: 1. Генеруємо випадковим чином початкову популяцію хромосом. 2. Оцінюємо придатність кожної хромосоми в популяції, вибираємо методом селекції пари хромосом. 3. Проводимо мутацію або ж схрещення з даними парами. 4. Покращуємо нашу популяцію завдяки хромосомам, які ми отримали на попередньому кроці. 5. Оновлюємо сукупність і копіюємо найкращого індивіда. Повторюємо кроки від 2 до 5 поки не знайдемо хорошої придатності, або не виконаються умови зупинки.

Чисельні результати

Розглянемо наступну одновимірну задачу,

$$T_t(x,t) = T_{XX}(x,t),\ 0 < x < 1,\ 0 < t < 1$$
$$T(x,0) = (\sin \pi x + \cos \pi x),\ 0 \leqslant x \leqslant 1$$
$$T(0,t) = e^{(-\pi^2 t)},\ 0 \leqslant t \leqslant 1$$
$$T(1,t) = q(t),\ 0 \leqslant t \leqslant 1$$

і умову перевизначеності

$$T(0.5, t_j) = s(t_j), t_j = j \times .05,\ j = 1,2,3, \dots ,20,$$

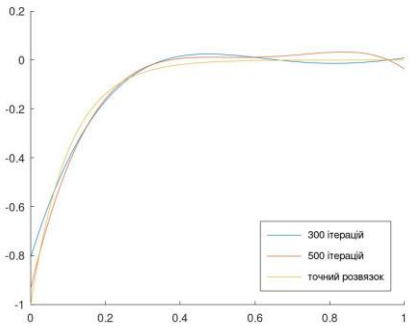
Під час втілення нашої програми ми розглянули 3 подання $q(t)$: $q(t) \approx g_1 * e^{g_2 * t}$, $q(t) \approx \frac{g_1}{(t+g_2)^{g_3}}$ та $q(t) \approx q_1 + q_2 t + q_3 t^2 + q_4 t^4 \dots + q_n t^n$.

За допомогою перших двох подань, ми можемо за короткий час найти дуже точні результати, але ці варінати є не універсальними. Розглянемо поліноміальне подання індивіду, яке ми можемо використати для розв'язання довільної задачі.

Запустимо програму з такими параметрами: $q(t)$ апроксимуємо поліномом 5 степеня. Популяція складається з 7000 індивідів. Програма виконає 300 та 500 ітерацій, для кожної з субпопуляції. Маємо 7 островів.

Ген1	Ген2	Ген3	Ген4	Ген5	Кількість ітерацій	Похибка $s(t)$
-0.808	4.924	-10.2669	8.816	-2.658	300	0.00191
-0.936	6.511	-16.462	18.062	-7.212	500	0.00097

Продемонструємо отримані результати для знаходження $q(t)$.



Висновок

В даній роботі було розглянуто застосування паралельних генетичних алгоритмів для оберненої задачі теплопровідності. Знаючи лише внутрішню температуру, ми змогли знайти відповідні граничні умови. Існують інші методи для розвязання цієї задачі, але на мою думку, метод генетичних алгоритмів чудово справився з поставленим перед ним завданням.

Аналізуючи чисельні експерименти, бачимо, що результат залежить від багатьох факторів, а саме: розміру популяції, подання індивіду, розмірності матриць при обчисленні прямої задачі теплопровідності, кількості ітерацій. Варто зазначити, що зроблена робота, дає нам можливість досить швидко та точно визначити граничні умови.

Список літератури

1. R. Pourgholi. Solving an inverse heat conduction problem using genetic algorithm: Sequential and multi-core parallelization approach / H. Dana, S. Tabasi. // Applied Mathematical Modelling. – 2014.
2. Z. Michalewicz. Genetic algorithm + data structures = evolutions programs // Springer. – 1996.
3. M. Raudenský. Genetic algorithm in solution of inverse heat conduction problems / A. Keith, J. Woodbury // An International Journal of Computation and Methodology. – Volume 28, 1995.
4. J. Koza. Genetic Programming // MIT Press, Cambridge. – 1990.

INVESTIGATION AND IMPLEMENTATION OF THE ITEM-BASED COLLABORATIVE FILTERING

Yaryna Paslavska

Ivan Franko National University of Lviv

Faculty of Applied Mathematics and Informatics

Yaryna.Paslavska@lnu.edu.ua

Collaborative filtering is one of the techniques used for recommending goods to people. It is based on gathering the information about clients' purchases or preferences in the system and using this data to recommend some new products to them. There are several types of collaborative filtering. Item-based collaborative filtering will be described and implemented in this work.

The main idea of this method is to find similar items to the items, bought or reviewed by each particular user. This includes several steps: comparing all the items between themselves, taking the user's favorite items, finding the most similar, and recommending them to the user. The first step is fundamental.

To begin with, the items comparison table is calculated based on the client's list of purchases or the client's reviews. The process requires several steps: finding users, who reviewed both items and based on their reviews calculating similarity using chosen metric. Pearson correlation coefficient was used for this investigation.

The significant advantage of item-based collaborative filtering is the fact that item similarity does not change rapidly or often. When the number of items or the number of reviews increases, similarity values do not change sharply. As a result, there is a possibility to store calculated similarities, use them when needed, and recalculate them occasionally when the server does not have a lot of work or even on another server.

The described algorithm was implemented using Python for calculations and MongoDB as data storage so that it is possible to integrate the solution into an e-commerce project in the future. A small example was created and tested for a better understanding of the algorithm from the beginning to the end. This example is described below. Supposing, there is a table with books and clients' rates for these books (image 1). The task is to recommend some items to a particular client.

	client #1	client #2	client #3	client #4	client #5	client #6	client #7	client #8	client #9	client #10
book #1	3		1		4	2			5	2
book #2		2	5				5	2		
book #3		3		4	4	1			3	1
book #4	5		5				3	2		
book #5	2		3		1	4			3	3
book #6	2	1		2				5		4
book #7		2			2	3	3		1	
book #8		3	4	5		1	2	3	4	
book #9	1		5		5			1		5
book #10		2		1	4	2	5		3	

Image 1. Table of books rates by client

Firstly, there is a need to calculate the similarity between each pair of books. For that, the Pearson correlation coefficient is used. Formula:

$$simil(x,y) = \frac{\sum_{i \in I_{xy}} (r_{x,i} - \overline{r_x})(r_{y,i} - \overline{r_y})}{\sqrt{\sum_{i \in I_{xy}} (r_{x,i} - \overline{r_x})^2 \sum_{i \in I_{xy}} (r_{y,i} - \overline{r_y})^2}},$$

where I_{xy} represents users, who reviewed both books. This formula returns result in the range $[-1, 1]$, but for easier calculation of the recommendations, the results were projected in the range $[0, 1]$. If a list of clients, who reviewed both books, is empty, the similarity is equal to 0. The results of the calculations are below.

	book1	book2	book3	book4	book5	book6	book7	book8	book9	book10
book1	1.000	0.000	0.926	0.000	0.281	0.000	0.009	0.639	0.371	0.827
book2	0.000	1.000	0.000	0.878	0.000	0.000	1.000	0.500	1.000	1.000
book3	0.926	0.000	1.000	0.000	0.081	0.107	0.176	0.985	0.000	0.590
book4	0.000	0.878	0.000	1.000	0.000	0.000	0.000	0.827	0.750	0.000
book5	0.281	0.000	0.081	0.000	1.000	1.000	0.664	0.000	0.587	0.009
book6	0.000	0.000	0.107	0.000	1.000	1.000	0.000	0.361	0.594	0.000
book7	0.009	1.000	0.176	0.000	0.664	0.000	1.000	0.028	0.000	0.592
book8	0.639	0.500	0.985	0.827	0.000	0.361	0.028	1.000	1.000	0.291
book9	0.371	1.000	0.000	0.750	0.587	0.594	0.000	1.000	1.000	0.000
book10	0.827	1.000	0.590	0.000	0.009	0.000	0.592	0.291	0.000	1.000

Image 2. Table of books similarities

Now there is the possibility to calculate recommendations for a particular client. For example, client #8. For this purpose, there is a need to gather information about already reviewed books in comparison to not reviewed books. In the table below, there is a comparison of already reviewed books to “new” books. There are such columns: titles of books that already have been reviewed, their rates, and similarity value to each “new” book (columns with the title “similarity book #..”). It is the value from the previous table. The last column for each book is expected rates for “new” books from client #8 (“expected rate ..”). It is the result of the multiplication of the similarity value and the client’s rate of the reviewed book. This is done to get an expected rate.

already reviewed books	rates from client #8	similarity book #1	expected rate for 1	similarity book #3	expected rate for 3	similarity book #5	expected rate for 5	similarity book #7	expected rate for 7	similarity book #10	expected rate for 10
book #2	2	0	0	0	0	0	0	1	2	1	2
book #4	2	0	0	0	0	0	0	0	0	0	0
book #6	5	0	0	0,107	0,535	1	5	0	0	0	0
book #8	3	0,639	1,917	0,985	2,955	0	0	0,028	0,084	0,291	0,873
book #9	1	0,371	0,371	0	0	0,587	0,587	0	0	0	0

Image 3. Table of expected rates for “new” books calculation

However, this table is just a preparation; now there is a need to calculate the possible client rates based on all reviewed books together. For this purpose, the sum of each column is calculated and then the final result is the division of the sum of the “expected rate” column by the sum of the “similarity book” column.

	0	1
0	book5	3.520
1	book3	3.196
2	book1	2.265
3	book10	2.225
4	book7	2.027

Image 4. Calculation results

In conclusion, the “new” books can be recommended to client #8 in such an order: book #5, book #3, book #1, book #10, and book #7. This is only a training example with a small number of items. When there are more items in the database, the system can pick the items with the highest expected rate.

As already mentioned, there is no need in updating the similarity table right away, so it can be used for some time and updated when it is convenient. Therefore, for the customer, there is just a need to compare his favorite products with others and find items with the highest expected rate. However, there is a need to take into consideration the memory costs of storing a similarity table.

In summary, item-based collaborative filtering is an efficient tool to calculate recommendations.

References

1. Alexandru Nedelcu Data mining finding similar items and users. [Electronic resource]. – Available from: https://alexn.org/blog/2012/01/16/cosine-similarity-euclidean-distance.html
2. Muffaddal Gutbuddin Comprehensive Guide on Item-Based Collaborative Filtering. [Electronic resource]. – Available from: https://towardsdatascience.com/comprehensive-guide-on-item-based-recommendation-systems-d67e40e2b75d
3. Toby Segaran Programming Collective Intelligence. – 2007, Vol. 2. – pp. 35-37

ПРОГРАМНО-АЛГОРИТМІЧНЕ ЗАБЕЗПЕЧЕННЯ ОПТИЧНОГО РОЗПІЗНАВАННЯ ТЕКСТУ ЗОБРАЖЕНЬ НИЗЬКОЇ ЯКОСТІ МЕТОДАМИ МАШИННОГО НАВЧАННЯ

Юліана Юрченко, Ярослав Соколовський

Львівський національний університет імені Івана Франка,

Факультет прикладної математики та інформатики
yuliana.yurchenko@lnu.edu.ua, yaroslav.i.sokolovskyi@lpnu.ua

Актуальність теми

Для використання найпопулярніших систем розпізнавання тексту рекомендується використовувати скануюче обладнання, яке є немобільним, часто недоступним або дорогим, а також непридатним до зчитування документу нестандартного формату (номерні знаки, білборди, інформаційні вивіски і т.д.). У повсякденному житті люди частіше використовують камери мобільних пристроїв та цифрові фотоапарати. Використання фотосенсорів і аналогової електроніки пристроями для реєстрації зображень неминуче призводить до появи спотворень зображень – цифрового шуму, який відображається на зображення у вигляді накладеної маски з пікселів випадкового кольору і яскравості. Цифровий шум посилюється в умовах недостатньої освітленості, внаслідок руху документа або камери під час експозиції.

Аналіз попередніх результатів

Серед традиційних методів оптичного розпізнавання тексту відомі численні висхідні підходи, де задача OCR виконується шляхом розв’язання незалежних дрібніших підзадач. Наприклад, для виявлення окремих символів користуються алгоритмами ковзаючого вікна [1, 2], зв’язаних компонент [3], голосування за Хафом [4] тощо. Виявлені символи інтегруються в слова за допомогою динамічного програмування, пошуку у лексиконі [1] тощо.

Інші роботи використовують наскрізні підходи, де текст розпізнається безпосередньо з цілого вхідного зображення. Наприклад, у праці [5] задачу розпізнавання тексту співставлено із задачею 90000-класової класифікації, де кожен клас визначає англійське слово. У роботі [6] побудовано CNN зі структурованим вихідним рівнем для розпізнавання тексту без сталого лексикону та без відомої довжини слова. Деякі роботи моделюють проблему OCR як проблему розпізнавання послідовності, а саме послідовності символів. Робота [7] вперше комбінує CNN та RNN, щоб отримати послідовні візуальні

ознаки заданого текстового зображення, а потім безпосередньо передати їх у декодер CTC.

Метою нашого дослідження є створення моделі оптичного розпізнавання тексту, яка ефективно працюватиме для зображень низької якості.

Створення набору даних

Для емулювання спотворень зображень тексту було створено набір зображень шляхом накладання тексту на білий фон; до згенерованих зображень застосовані функції зміни розмірів, функції накладання шуму, оператори розмиття та повороту. Усі застосовані спотворення мають випадковий рівень інтесивності. Створення даних у такий спосіб дало змогу отримати розмічені зображення тексту низької якості у будь-якій кількості.

Архітектура нейронної мережі

З використанням бібліотеки Keras побудовано архітектуру CRNN, для якої визначена функція втрат CTC. Для збільшення точності результатів OCR у цій роботі запропоновані етапи попередньої обробки даних: алгоритм горизонтального вирівнювання зображення тексту та алгоритм розрізання зображення багаторядкового тексту на декілька зображень однорядкового тексту.

Алгоритм горизонтального вирівнювання зображення тексту

Зображення приводимо до вигляду одноканального у відтінках сірого.

Позначимо значення пікселя як $p_{i,j} \in [0;256)$, $\overline{i = 1,n}$, $\overline{j = 1,m}$; n, m – кількості рядків та стовпців зображення відповідно.

Нехай тоді x = $\{x_1,x_2,...,x_n\}$, де

$$x_i = \sum_{j=1}^{m}(1 - \frac{p_{i,j}}{255}), \quad i = \overline{1,n}, \qquad (1)$$

чисельно виражає кількість чорного кольору в кожному рядку.

Для горизонтального вирівнювання зображення повернемо його на кут ϕ, для якого у результуючого зображення коефіцієнт варіації ознаки x є максимальним.

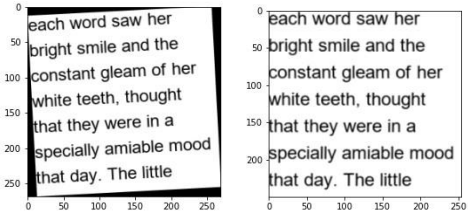


Рис. 1: Результат встановлення горизонтального положення зображення

Алгоритм розрізання зображення багаторядкового тексту на декілька зображень однорядкового тексту

Тепер визначимо місця можливих розрізань зображення після рядків *i*, для яких:

- $\{x_{i-\alpha},\ldots,x_{i-2},x_{i-1}\}$ утворюють область низької кількості чорного кольору. Визначимо це так:

$$\overline{x_\alpha} = \frac{1}{\alpha}\sum_{j=i-\alpha}^{i-1} x_j \quad < \quad \overline{x} - \frac{1}{2}\sigma \tag{2}$$

- $\{x_i,x_{i+1},\ldots,x_{i+\beta}\}$ утворюють область високого значення приросту кількості чорного кольору. Визначимо це так:

$$\phi_\beta = \frac{\sum_{j=i}^{i+\beta-1} x_j - \sum_{j=i-\beta}^{i-1} x_j}{\beta} \quad > \quad \delta \cdot \sigma \tag{3}$$

Тут α, β та δ – регульовані гіперпараметри. У роботі ці гіперпараметри були визначенні як α = 0.04, β = 0.02, δ = 0.3. На Рис. 2 місця можливих розрізань позначені червоним кольором. З кожної групи таких індексів вибирається перший (лівий край на Рис. 2 позначений пунктиром).

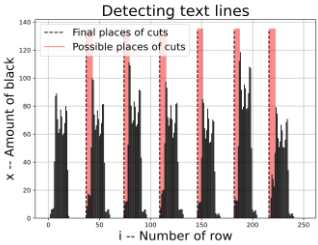


Рис. 2: Графік ознаки x та результів виявлення текстових рядків зображення

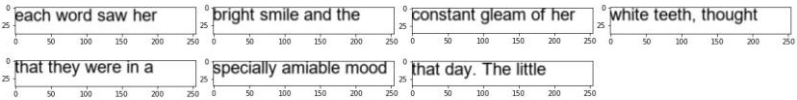


Рис. 3: Результат роботи алгоритму розрізання зображення багаторядкового тексту на декілька зображень однорядкового тексту

Результати ефективності моделі

Дані для тренування, валідації та тестування поділені у відношенні 9 : 1 : 1 зображень відповідно.

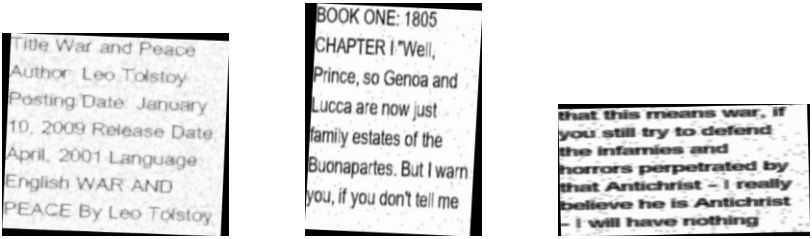


Рис. 4: Зразки набору даних зображень низької якості

Як метрику порівняння взято відношення відстані Левенштейна [10] між передбаченням моделі та міткою до довжини мітки. Для порівняння використовували Tesseract OCR v5.0.0. (alpha) [8] та Google Cloud Vision API [9] (останній тільки для 50 зображень). Значення метрики порівняння наведені в Таблиці 1.

Табл. 1: Середні значення метрики порівняння на тестовому наборі даних для нашої моделі, Tesseract та Google Cloud Vision API

	Наша модель	Tesseract	Cloud Vision API
50 тестових зображень	0.022326	0.742395	0.190283
1000 тестових зображень	0.019783	0.726493	–

Висновок

Отже, запропонований нами алгоритм дає високу точність розпізнавання тексту на створеному наборі даних зображень низької якості.

Використана література:

1. Wang K., Babenko B., and Belongie S. *End-to-end scene text recognition*. In ICCV, 2011.
2. Wang K. and Belongie S. *Word spotting in the wild*. In ECCV, 2010.
3. Neumann L. and Matas J. *Real-time scene text localization and recognition*. In CVPR, 2012.
4. Yao C., Bai X., Shi B., and Liu W. *Strokelets: A learned multi-scale representation for scene text recognition*. In CVPR, 2014.
5. Jaderberg M., Simonyan K., Vedaldi A., and Zisserman A. *Reading text in the wild with convolutional neural networks*. Int. J. Comput. Vision, 2015.

6. Jaderberg M., Simonyan K., Vedaldi A., and Zisserman A. *Deep structured output learning for unconstrained text recognition*. In ICLR, 2015.
7. Shi B., Bai X., Yao C. *An End-to-End Trainable Neural Network for Imagebased Sequence Recognition and Its Application to Scene Text Recognition*.
8. GitHub repository for Tesseract OCR v5.0.0 (alpha) https://github.com/
9. UB-Mannheim/tesseract/wiki
10. *Google Cloud Vision API* – Available from: https://cloud.google.com/ vision
11. Левенштейн В. И., *Двоичные коды с исправлением выпадений, вставок и замещений символов*, Докл. АН СССР, 163, 4, 1965, 845-848

ЗАСТОСУВАННЯ ПАРАЛЕЛЬНИХ СХЕМ ПРИ РОЗВ'ЯЗУВАННІ СИСТЕМ МАТРИЧНИХ РІВНЯНЬ

Любомир Фадєєв

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

l1.fadieiev@gmail.com

Вступ

Матричні рівняння та системи матричних рівнянь широко використовуються у задачах оптимізації систем управління. Існують різні методи розв'язування подібних задач, проте всі вони є послідовними. Через це час їхнього виконання може бути досить тривалим, при цьому час виконання послідовних алгоритмів росте зі збільшенням розмірності матриць у рівняннях.

Цю проблему частково вирішує застосування у послідовних алгоритмах паралельних схем. Найбільшої ефективності ми досягнемо, якщо застосуємо схеми до найбільш тривалих дій, як-от множення матриць, обчислювальна складність якого може досягати $O(n^3)$. Це дозволяє значно зменшити час розв'язування матричних рівнянь, а також повною мірою скористатися можливостями сучасних обчислювальних машин, більшість з яких здатна виконувати алгоритми паралельно.

Отже, актуальність використання паралелізму при розв'язуванні матричних рівнянь і систем матричних рівнянь є очевидною. В даній роботі розглянемо розв'язування системи матричних рівнянь з використанням паралельних схем при обчисленні добутку двох матриць.

Постановка задачі

Розглядаємо систему з n матричних рівнянь другого степеня:

$$\begin{cases} A_{1,X_1X_1}X_1^2 + A_{1,X_1X_2}X_1X_2 + \ldots + A_{1,X_1X_n}X_1X_n + A_{1,X_2X_1}X_2X_1 + \\ +A_{1,X_2X_n}X_2X_n + \ldots + A_{1,X_nX_n}X_n^2 + A_{1,X_1}X_1 + A_{1,X_2}X_2 + \ldots + \\ +A_{1,X_n}X_n + A_{1,1} = 0; \\ A_{2,X_1X_1}X_1^2 + A_{2,X_1X_2}X_1X_2 + \ldots + A_{2,X_1X_n}X_1X_n + A_{2,X_2X_1}X_2X_1 + \\ +A_{2,X_2X_n}X_2X_n + \ldots + A_{2,X_nX_n}X_n^2 + A_{2,X_1}X_1 + A_{2,X_2}X_2 + \ldots + \\ +A_{2,X_n}X_n + A_{2,1} = 0; \\ \vdots \\ A_{n,X_1X_1}X_1^2 + A_{n,X_1X_2}X_1X_2 + \ldots + A_{n,X_1X_n}X_1X_n + A_{n,X_2X_1}X_2X_1 + \\ +A_{n,X_2X_n}X_2X_n + \ldots + A_{n,X_nX_n}X_n^2 + A_{n,X_1}X_1 + A_{n,X_2}X_2 + \ldots + \\ +A_{n,X_n}X_n + A_{n,1} = 0; \end{cases}$$

де невідомі $X_i (i = 1,2,\ldots,n)$ і коефіцієнти $A_{l,X_i,X_j}, A_{l,X_i} (l = 1,\ldots,n;\ i,j = 1,2,\ldots,n)$ є дійсними квадратними некомутативними матрицями розмірності $m \times m$.

Для знаходження розв'язку системи рівнянь (1) можемо описати наступний алгоритм:

1. Задаємо похибку $\varepsilon > 0$;
2. Задаємо початкове наближення алгоритму – дійсні квадратні матриці $X_i^{(0)} (i,j = 1,2,\ldots,n)$;
3. Встановлюємо лічильник k = 1;
4. Обчислюємо $X_i^{(1)} (i = 1,2,\ldots,n)$ за формулою:

$$\begin{pmatrix} X_1^{(1)} \\ X_2^{(1)} \\ \vdots \\ X_n^{(1)} \end{pmatrix} = \left(\begin{pmatrix} B_1 & B_2 & \ldots & B_n \end{pmatrix} \circ \begin{pmatrix} X_1^{(0)} \\ X_2^{(0)} \\ \vdots \\ X_n^{(0)} \end{pmatrix} + B_{n+1} \right)^{-1} Y.$$

де B_i – матриця коефіцієнтів перед матрицею X_i , $Y = -(A_{1,1}\, A_{2,1} \ldots A_{n,1})^T$.

1. Перевіряємо умову $\left\| X_i^{(1)} - X_i^{(0)} \right\| \leq \varepsilon\ (i = 1,2,\ldots,n)$. Якщо умова не виконується, встановлюємо лічильник $k = k + 1$ присвоюємо матрицям $X_i^{(0)}$ значення матриць $X_i^{(1)} (i = 1,2,\ldots,n)$ і переходимо до кроку 4, інакше повертаємо $X_i^{(1)} (i = 1,2,\ldots,n)$.

При виконанні алгоритму дія множення матриць виконується досить часто ($n + 1$ разів кожного кроку), що робить цей метод дуже ресурсозатратиним. Щоб зменшити час виконання програми, ми скористаємося властивістю множення матриць: його можна подати у вигляді скалярних добутків між рядками першої та стовпцями другої матриці. При цьому пошук кожного скалярного добутку є незалежним від інших, а отже степінь паралелізму (кількість операцій, які можна здійснити паралельно) алгоритму – m^2 (де $n \times m$ та $m \times k$ – розмірність першої та другої матриці відповідно).

Чисельні експерименти

Програмна реалізація втілена на мові C# за допомогою Visual Studio. Для більш ефективної реалізації паралельних схем було застосовано бібліотеку TPL (Task Parallel Library), а саме клас Parallel.

Розглянемо систему (1). За алгоритмом (2) будемо обчислювати розв'язок для різних k (кількості кроків алгоритму) та n (розмірність квадратних матриць з системи (1)).

Табл 1. Результати обчислень для системи (1).

k	n	Час виконання послідовного алгоритму, мс	Час виконання алгоритму із застосуванням схеми паралельного виконання множення матриць, мс
100	50x50	9386	8852
100	60x60	24672	14709
100	70x70	43432	24100
200	50x50	42471	17540
200	60x60	48963	31708
200	70x70	89164	50261
300	50x50	77837	25888
300	60x60	72937	45640
300	70x70	131620	76058

Висновки

У цьому дослідженні було розглянуто перспективу розробки паралельних алгоритмів для обчислення матричних рівнянь та систем матричних рівнянь. Також були проведені чисельні експерименти, які детальніше будуть продемонстровані у доповіді.

Список літератури

[1] James M. Ortega Introduction to Parallel and Vector Solution of Linear Systems / James M. Ortega. – Plenum Press New York and London, 1988. – 305с.

[2] Недашковська А.М. Ітераційний метод розв'язування системи поліноміальних рівнянь другого степеня// Фізико-математичне моделювання та інформаційні технології. -2015р. -Вип. 21. С.150-161.

ПРОЕКТУВАННЯ ТА РОЗРОБКА МОБІЛЬНОГО АСИСТЕНТА "WINGMAN"

Андрій Бурцьо

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

burco.ab@gmail.com

Вступ

В сучасному світі мобільний телефон став невід'ємною частиною нашого життя, проте окрім телефону користувачі щоденно взаємодіють із іншими пристроями такими як ноутбук, планшет чи персональний комп’ютер. І нажаль не всі мають свою еко систему із голосовим асистентом як Siri в Apple.

Wingman - це мобільний асистент який має скласти конкуренцію Apple Siri, Amazon Alexa, Microsoft Cortana та іншим, у спрощенні взаємодії з великою кількістю пристроїв та об'єднанні їх в зручну екосистему.

Архітектура та використані технології

При проектуванні мобільного асистента "Wingman" за основу була взята Onion архітектура [1]. Концепція даного підходу базується на чотирьох основних рівнях: рівень об’єктів домену, рівень сховища, рівень сервісу, рівень інтерфейсу користувача. В даної архітектури є низка переваг зокрема вона є легкою в підтримці та тестуванні, а також через те що за допомогою цієї архітектури створюється слабко пов’язана програма, оскільки зовнішній рівень програми завжди взаємодіє з внутрішнім шаром через інтерфейси, спрощується реалізація мобільного асистента на різних платформах.

Для реалізації даного проекту було використано низку сучасних рішень. Зокрема для реалізації серверної частини було взято за основу ASP.NET Core Web API [2], в якості ORM використовувався Entity Framework Core із підходом Code-First. В якості сховища данних було використано полегшену реляційну систему керування базами даних SQLite. З метою впровадження системи автентифікації користувачів, а також можливості їх авторизації, було використано платформу Microsoft Identity [3] та стандарт JWT (Json Web Token) [4]. Також було реалізовано підтримку автентифікації за допомогою облікових даних Google, із допомогою платформи Firebase, що в подальшому

полегшить взаємодією із Google додатками зокрема на платформі Android. Для реалізації рішення на Android, IOS та Windows використовувалась платформа розробки Xamarin [5], а для коректної взаємодії мобільного асистента на усіх вище наведених платформах було використано набір клієнтських бібліотек WCF [6]. Оскільки управління голосовим асистентом відбувається голосовими командами то в якості сервісу розпізнавання голосу було використано Azure Speech Service [7].

Можливості мобільного асистента

Основним функціоналом даного мобільного асистента є віддалений контроль усіх підключених користувачем пристроїв. Наприклад виключення ноутбука за допомогою телефона, чи використовуючи екранну клавіатуру планшета набирати текст на комп'ютері. Також у ньому присутній функціонал такий як налаштування будильника, додавання нагадувань та додавання подій в календар та інший функціонал, який дозволяє взаємодіяти з технологіями Інтернет речей для зручного без контактного управління домом.

Висновок

Спроектовано та реалізовано мобільний асистент "Wingman", що дозволяє користувачам зручно працювати із багатьма пристроями на платформах таких як Android, IOS та Windows. Завдяки мобільному асистенту користувач системи швидко та без зайвих зусиль має можливість видобути корисну та важливу інформацію.

Література

1. Onion Architecture In ASP.NET Core MVC [Електронний ресурс] – Режим доступу до ресурсу: https://www.c-sharpcorner.com/article/onion-architecture-in-asp-net-core-mvc/
2. ASP.NET Core [Електронний ресурс] – Режим доступу до ресурсу: https://app.pluralsight.com/paths/skill/aspnet-core
3. Microsoft Identity [Електронний ресурс] – Режим доступу до ресурсу: https://aspnetcore.readthedocs.io/en/stable/security/authentication/identity.html
4. JSON Web Tokens [Електронний ресурс] – Режим доступу до ресурсу: https://jwt.io/.
5. What is Xamarin? [Електронний ресурс] – Режим доступу до ресурсу: https://docs.microsoft.com/en-us/xamarin/get-started/what-is-xamarin
6. What Is Windows Communication Foundation [Електронний ресурс] – Режим доступу до ресурсу: https://docs.microsoft.com/en-us/dotnet/framework/wcf/whats-wcf
7. Speech recognition using Azure Speech Service [Електронний ресурс] – Режим доступу до ресурсу: https://docs.microsoft.com/en-us/xamarin/xamarin-forms/data-cloud/azure-cognitive-services/speech-recognition

РОЗПІЗНАВАННЯ ОБ'ЄКТІВ У РЕАЛЬНОМУ ЧАСІ ЗА ДОПОМОГОЮ TENSORFLOW У МОБІЛЬНИХ ДОДАТКАХ

Станіслав Чернецький

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

stanislav.chernetskyi@lnu.edu.ua

Вступ

Розпізнавання об'єктів за допомогою TensorFlow – це техніка комп'ютерного зору. TensorFlow допомагає нам виявляти, знаходити та відстежувати об'єкт із зображення чи відео.

Що таке розпізнавання об'єктів

Розпізнавання об'єктів — це техніка комп'ютерного зору, за допомогою якої програмна система може виявляти, знаходити та відстежувати об'єкт із заданого зображення чи відео. Особливим атрибутом виявлення об'єктів є те, що він ідентифікує клас об'єкта (людина, стіл, стілець тощо) та їхні координати на даному зображенні. Місце розташування вказується шляхом малювання обмежувальної рамки навколо об'єкта. Обмежувальна рамка може точно визначити його положення. Можливість знаходити об'єкт всередині зображення визначає продуктивність алгоритму, який використовується для виявлення. Розпізнавання обличчя є одним із прикладів виявлення об'єктів.

Ці алгоритми виявлення об'єктів можуть бути попередньо навчені або їх можна навчити з нуля. У більшості випадків ми використовуємо алгоритми з попередньо навченю моделлю, а потім точно налаштовуємо їх відповідно до наших вимог і різних випадків.

Використання фреймворку для розробки додатків під IOs і Android

Ми розглянемо різницю між двома провідними технологіями - Flutter і React Native.

Flutter - це портативний набір інструментів інтерфейсу користувача для створення власно скомпільованих додатків для мобільних пристроїв, веб сторінок та комп'ютерних додатків з єдиною кодовою базою. Розроблений фреймворк компанією Google, використовує мову Dart.

React Native - це фреймворк для створення нативних програм за допомогою React. Розроблений компанією Facebook і використовує мову JavaScript.

Обидві технології мають як плюси, так і мінуси. Ми будемо порівнювати продуктивність, досвід розробки, різницю між Dart і JavaScript, стандартні бібліотеки та багато іншого.

Важко порівнювати продуктивність будь-якого фреймворку. Flutter і React Native достатньо швидкі для більшості ситуацій. Історично продуктивність Flutter була кращою ніж у React Native. Різниця продуктивності в основному пов'язана з асинхронним мостом між native code і JavaScript. Flutter використовує відому графічну бібліотеку Skia.

Налаштувати середовище для розробки Flutter, як правило, легше, ніж React Native. Hot Reload у Flutter зазвичай працює краще, ніж Fast Refresh у React Native. У них є дуже гарні інструменти для debugging, profiling та роботи з деревом віджетів. Їхнє вбудоване end-to-end тестування набагато краще, ніж у React Native Detox.

Flutter компілює код Dart на нативний і має досить круту модель інтеграції з рідним кодом, використовуючи Platform Channels. Він дозволяє виконувати синхронні вихідні виклики та дозволяє писати нативний код на Swift та Kotlin.

У React Native інтеграція з native вимагає певного досвіду. Не кажучи вже про те, щоб боротися з обмеженнями React Native bridge. У зв'язку з цими перевагами я вибрав Flutter для своєї роботи.

Як працює розпізнавання об'єктів?

Як правило, розпізнавання об'єкта виконується в три етапи:

1. Генерує маленькі сегменти у вхідних даних.
2. Вилучення ознак виконується для кожної сегментованої прямокутної області, щоб передбачити, чи містить прямокутник дійсний об'єкт.
3. Прямокутники що перекриваються, об'єднуються в єдиний обмежуючий прямокутник.

Що таке TensorFlow?

Tensorflow – це бібліотека з відкритим вихідним кодом для чисельних обчислень і велико-масштабного машинного навчання.

Tensorflow об'єднує моделі та алгоритми машинного навчання та глибокого навчання.

Він використовує Python як зручний інтерфейс і ефективно запускає його на оптимізованому C++.

Tensorflow дозволяє розробникам створювати графік обчислень для виконання.

Кожен вузол на графіку представляє математичну операцію, а кожне з'єднання – дані. Таким чином, замість того, щоб мати справу з низькими деталями, наприклад, з'ясовувати правильні способи прив'язати вихід однієї функції до входу іншої, розробник може зосередитися на загальній логіці програми.

TensorFlow на даний момент є найпопулярнішою бібліотекою програмного забезпечення. Існує кілька реальних застосувань глибокого навчання, які роблять TensorFlow популярним. Будучи бібліотекою з відкритим вихідним кодом для глибокого навчання та машинного навчання, TensorFlow відіграє свою роль у текстових програмах, розпізнаванні зображень, голосовому пошуку та багато іншого. DeepFace, система розпізнавання зображень Facebook, використовує TensorFlow для розпізнавання зображень. Його використовує Siri від Apple для розпізнавання голосу. Кожен додаток Google, який ви використовуєте, зазвичай використовує TensorFlow.

Результат

Розроблений додаток за допомогою фреймворка Flutter для розпізнавання об'єктів за допомогою TensorFlow

Література

1. Real Time Object Detection and Recognition using Deep Learning Methods// Sai Krishna Chadalawada

2. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Uni- fied, real-time object detection," in Proceedings of the IEEE conference on com- puter vision and pattern recognition, 2016, pp. 779–788.

3. C. C. Nguyen, G. S. Tran, T. P. Nghiem, N. Q. Doan, D. Gratadour, J. C. Burie, and C. M. Luong, "Towards real-time smile detection based on faster region convolutional neural network," in 2018 1st International Conference on Multimedia Analysis and Pattern Recognition (MAPR). IEEE, 2018, pp. 1–6.

4. Z. Deng, H. Sun, S. Zhou, J. Zhao, L. Lei, and H. Zou, "Multi-scale object detection in remote sensing imagery with convolutional neural networks," ISPRS journal of photogrammetry and remote sensing, vol. 145, pp. 3–22, 2018.

5. D. Xiao, F. Shan, Z. Li, B. T. Le, X. Liu, and X. Li, "A target detection model based on improved tiny-yolov3 under the environment of mining truck," IEEE Access, vol. 7, pp. 123 757–123 764, 2019.

6. Flutter site. [Online]. Available: https://flutter.dev/

7. React Native site. [Online]. Available: https://reactnative.dev/

MATHEMATICAL MODELING OF ACOUSTIC WAVE PROPAGATION IN VISCOUS FLUID

Anastasiia Bondarenko, Vitaliy Horlatch

Ivan Franko National University of Lviv

Faculty of Applied Mathematics and Informatics

anastasiia.bondarenko@lnu.edu.ua

Introduction

The problem of dissipative acoustics have many applications in engineering, medicine, life safety, etc. For instance, analysis of underwater structures on the basis of the echo signal, ultrasound diagnostics, noise reduction and much more. This paper presents mathematical model and numerical results of acoustic wave propagation in viscous fluid in terms of unknown displacements.

Initial-boundary problem

Let the fluid occupy limited area Ω of points $x=(x_1,x_2,x_3)$ of Euclidean space $\mathbf{R}^3$, and let the area Ω has Lipschitz-continuous boundary Γ. Let us introduce piecewise defined displacement vector $\mathbf{u}(x,t)=\{u_i(x,t)\}_{i=1}^3$ for all $(x,t)\in\Omega\times[0,T]$ which describes the motion of the fluid and satisfies the following equation

$$\rho u_i'' - \sigma_{ij,j} = 0,$$

$$\sigma_{ij}(\mathbf{u}) = a_{ijkm}e_{km}(\mathbf{u}) + c_{ijkm}e_{km}(\mathbf{u}'),$$

$$e_{ij}(\mathbf{u}) = (u_{i,j}+u_{j,i})/2 \quad in \quad \Omega\times(0,T],$$

as well as boundary and initial conditions

$$\begin{cases} u_i = 0 & on \quad \Gamma_U\times[0,T], \quad \Gamma_U\subset\Gamma, \quad mes(\Gamma_U)>0, \\ \sigma_{ij}\nu_j = \hat{\sigma}_i & on \quad \Gamma_\sigma\times[0,T], \quad \Gamma_\sigma = \Gamma\setminus\Gamma_U. \end{cases}$$

$$\begin{cases} \mathbf{u}_{t=0} = \mathbf{u}^0, \\ \mathbf{u}'_{t=0} = \mathbf{u}^1 \end{cases} \quad in\ \Omega$$

Here $\rho=\rho(x)$ is the density, $a_{ijkm}=a_{ijkm}(\mathbf{x})$ and $c_{ijkm}=c_{ijkm}(\mathbf{x})$ are modules of elasticity and viscosity respectively, $\hat{\sigma}_i=\hat{\sigma}_i(\mathbf{x},t)$, $\mathbf{u}^0=\mathbf{u}^0(\mathbf{x})$ and $\mathbf{u}^1=\mathbf{u}^1(\mathbf{x})$ are given functions and $\mathbf{v}=\{\nu_i(\mathbf{x})\}_{i=1}^3$ is a unit vector of the outer normal to the boundary Γ.

Variational formulation

Let us consider the space of admissible displacements $V = \{\mathbf{v} \in [H^1(\Omega)]^3 \,|\, \mathbf{v} = 0 \ on\ \Gamma_U\}$ and $H = [L^2(\Omega)]^3$ and formulate the variational problem [1]

$$Let \quad \mathbf{l} \in L^2(0,T;V'), \mathbf{u}^0 \in V, \mathbf{u}^1 \in H;$$

$$find\ \ vector\ \ \mathbf{u} \in L^2(0,T;V) \cap L^\infty(0,T;H),\ that$$

$$m(\mathbf{u}''(t),\mathbf{v}) + c(\mathbf{u}'(t),\mathbf{v}) + a(\mathbf{u}(t),\mathbf{v}) = <\mathbf{l}(t),\mathbf{v}> \quad \forall t \in (0,T],$$

$$a(\mathbf{u}(0) - \mathbf{u}^0, \mathbf{v}) = 0, \quad m(\mathbf{u}'(0) - \mathbf{u}^1, \mathbf{v}) = 0 \qquad \forall \mathbf{v} \in V.$$

where bilinear forms and linear functional listed above have the following form:

$$\begin{cases} m(\mathbf{u},\mathbf{v}) = \int\limits_\Omega \rho u_i v_i \,\mathrm{d}\mathbf{x}, \qquad a(\mathbf{u},\mathbf{v}) = \int\limits_\Omega a_{ijkm} e_{ij}(\mathbf{u}) e_{ij}(\mathbf{v}) \,\mathrm{d}\mathbf{x}, \\ c(\mathbf{u},\mathbf{v}) = \int\limits_\Omega c_{ijkm} e_{ij}(\mathbf{u}) e_{ij}(\mathbf{v}) \,\mathrm{d}\mathbf{x} \quad \forall \mathbf{u},\mathbf{v} \in V, \\ <\mathbf{l},\mathbf{v}> = \int\limits_{\Gamma_\sigma} \hat{\sigma}_i v_i \,\mathrm{d}\gamma \quad \forall \mathbf{v} \in V. \end{cases}$$

Energy balance equation

The energy balance equation of this problem has the following structure [3]

$$\tfrac{1}{2}[\|\mathbf{u}'(t)\|_H^2 + |\mathbf{u}(t)|_V^2] + \int_0^t \|\mathbf{u}'(\tau)\|_V^2 \, d\tau$$

$$= \tfrac{1}{2}[\|\mathbf{u}^1\|_H^2 + |\mathbf{u}^0|_V^2] + \int_0^t [<\mathbf{l}(\tau),\mathbf{u}'(\tau)>] d\tau \quad \forall t \in [0,T].$$

Galerkin semi discretization

Let $\{V_h\}$ be a sequence of finite-dimensional spaces such that $V_h \subset V$, $\dim V_h = N = N(h) \to \infty$ when $h \to 0$. Moreover $\bigcup\limits_{h>0} V_h$ is dense in V. Also we choose basic functions

$$\mathbf{u}_h(\mathbf{x},t) = \sum_{k=1}^N U_k(t)\varphi_k(\mathbf{x}).$$

Substituting these functions into variational formulation, we obtain the following Cauchy problem in matrix form:

Find $\mathbf{U}(t) = \{U_i(t)\}_{i=1}^N$ such that

$$\begin{cases} \mathbf{MU}''(t) + \mathbf{CU}'(t) + \mathbf{AU}(t) = \mathbf{L}(t) \qquad t \in (0,T], \\ \mathbf{AU}(0) = \mathbf{U}^0 , \\ \mathbf{MU}'(0) = \mathbf{U}^1 . \end{cases}$$

where matrices **M, C, A** and vector **L** are calculated by the expressions for bilinear forms and linear functional.

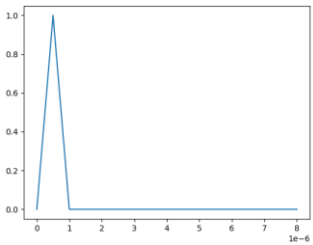
One-step recurrent scheme of Cauchy integration over time

Based on the Petrov-Galerkin procedure [1] with stepwise by quadratic displacement approximation by a complete analogy with [2] we obtain the one-step recurrent scheme (ORS) of Cauchy integration over time.

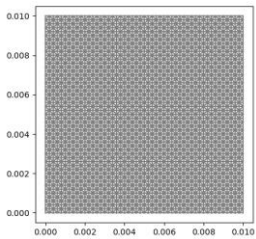
$$\begin{cases} Let\ \mathbf{U}^0, \dot{\mathbf{U}}^0, \Delta t > 0, \quad \gamma, \beta \in [0,1]. \\ Find\ \ddot{\mathbf{U}}^{j+1/2}\ and\ \mathbf{U}^{j+1}, \dot{\mathbf{U}}^{j+1}, \quad that \\ \left[\mathbf{M} + \Delta t\,\gamma\,\mathbf{C} + \frac{1}{2}\Delta t^2 \beta\,\mathbf{A}\right]\ddot{\mathbf{U}}^{j+1/2} = \mathbf{L}\left(t_{j+1/2}\right) - \mathbf{C}\dot{\mathbf{U}}^j - \mathbf{A}\left[\mathbf{U}^j + \Delta t\,\gamma\,\dot{\mathbf{U}}^j\right], \\ \dot{\mathbf{U}}^{j+1} = \dot{\mathbf{U}}^j + \Delta t\,\ddot{\mathbf{U}}^{j+1/2}, \\ \mathbf{U}^{j+1} = \mathbf{U}^j + \frac{1}{2}\Delta t\left(\dot{\mathbf{U}}^j + \dot{\mathbf{U}}^{j+1}\right), \\ j = 0,1,\ldots\ while\ \ j\Delta t \le T. \end{cases}$$

Numerical experiments

All experiments were performed using FEniCS Project [4,5]. For our experiments, consider a square-shaped area with sides of 0.01m, which is filled with glycerin with physical quantities $\rho = 1260, c = 1920, \eta = \xi = 1.49$. On the right side of the square ($x = 0.01$) the boundary conditions are $\hat{\boldsymbol{\sigma}}(t) = (\sigma^*(t), 0)$ and on other sides zero displacements $u_x = u_y = 0$ are specified. The step integration over time is $\Delta t = 5\times10^{-9} s$ and $T = 8\times10^{-6} s$.



$\sigma^*(t)$



50x50 triangular mesh

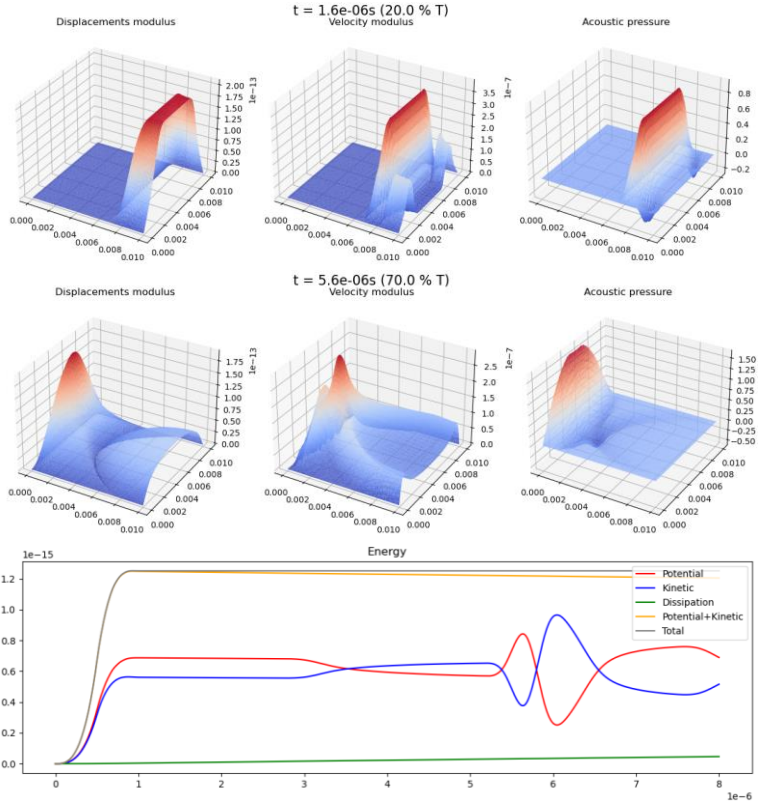


Figure 1. Wave propagation

From the obtained results we can say that surfaces of solutions are smooth, what is expected for glycerin. The conservation of energy law is fulfilled.

References

1. Shynkarenko H. A. Projection-mesh methods for solving initial boundary value problems / H. A. Shynkarenko – KA: SMC IVO, 1991. – 88 p. (in Ukrainian).
2. Horlatch V. Construction and analysis of one-step recurrent scheme for time integration of a variational dissipative acoustics problem / V. Horlatch, I. Klymenko, H. Shynkarenko // Visnyk of the Lviv University. Series Appl. Math. And Informatics – 2012. № 18. – pp. 76-95. (in Ukrainian).
3. Horlatch V. Formulation and well-posedness of the variational problem of viscous heat-conducting fluid acoustics / V. Horlatch, I. Klymenko and G. Shynkarenko // Journal of Numerical and Applied Mathematics – 2012. № 3 (109) – pp. 53-71.
4. The FEniCSx computing platform. Available from: [https://fenicsproject.org/]
5. Logg A. Automated Solution of Differential Equations by the Finite Element Method. The FEniCS Book / A. Logg, K.-A. Mardal, G. Wells. – Springer, 2012. – 723 p.

MATHEMATICAL MODELING AND ANALYSIS OF STATIONARY ACOUSTIC WAVE PROPAGATION IN VISCOUS FLUID

Volodymyr Milchanovskyi, Vitaliy Horlatch

Ivan Franko National University of Lviv

Faculty of Applied Mathematics and Informatics

volodymyr.milchanovskyi@lnu.edu.ua

Introduction

The study of acoustic wave propagation processes is one of the important objectives in many fields. The complexity of the acoustic wave structure requires the use of modern computer modeling methods with advanced numerical techniques, the analysis of the stability and convergence of which is also an important problem. This paper presents mathematical model and analysis of stationary acoustic wave propagation in viscous fluid.

Initial-boundary problem

Let fluid occupy limited area Ω of points $x = \{x_j\}_{j=1}^{d} \in R^d, d = 1, 2 \, or \, 3$, and let area Ω has Lipschitz-continuous boundary Γ. To describe the motion of the fluid we present piecewise defined displacement vector $\boldsymbol{u} = \{u_j(x,t)\}_{j=1}^{d}$ for all $(x,t) \in \Omega \times [0, \mathrm{T}]$ that satisfies the following equation

$$\begin{cases} \rho u_i'' - \sigma_{ij,j} = 0, \\ \sigma_{ij}(u) = a_{ijkm} e_{km}(u) + c_{ijkm} e_{km}(u'), \\ e_{ij}(u) = \dfrac{(u_{i,j} + u_{j,i})}{2} \, on \, \Omega \in (0,T] \end{cases}$$

with boundary conditions

$$\begin{cases} u_j = 0 \, on \, \Gamma_U \times [0,T], \Gamma_U \subset \Gamma, mes(\Gamma_U) > 0, \\ \sigma_{ij} \nu_j = \hat{\sigma}_i \, on \, \Gamma_\sigma \times [0,T], \Gamma_\sigma = \Gamma \setminus \Gamma_U \end{cases}$$

and initial conditions

$$\begin{cases} u_{t=0} = u^0, \\ u'_{t=0} = u^1 \end{cases} on \, \Omega$$

where ρ – density, a_{ijkm} – elasticity, c_{ijkm} – viscosity, $u^0(x), u^1(x), \hat{\sigma}_i(x,t)$ – given functions and $v=\{v_i(x)\}_{i=1}^{d}$ is unit vector of the outer normal to the boundary Γ.

Variational formulation

Consider the case when the oscillations in the system are caused by boundary conditions on Γ_σ that change in time according to a harmonic law with a given circular frequency of oscillations $\omega > 0$, so that the function $\boldsymbol{l}(t)$ in the right side of equation takes the form $\boldsymbol{l}(t)=\boldsymbol{l}^1\sin(\omega t)+\boldsymbol{l}^2\cos(\omega t)$.

Then we can assume that after a certain long period of time the influence of the initial conditions will be negated, and the fluctuations in the system will occur according to the same harmonic law $\boldsymbol{u}(x,t)=\boldsymbol{u}^1\sin(\omega t)+\boldsymbol{u}^2\cos(\omega t)$, where $\boldsymbol{u}^1=\boldsymbol{u}^1(x), \boldsymbol{u}^2=\boldsymbol{u}^2(x)$ – unknown amplitudes of acoustic displacement.

Then we have

$$\boldsymbol{u}'(x,t)=\omega\boldsymbol{u}^1(x)\cos(\omega t)-\omega\boldsymbol{u}^2\sin(\omega t)$$

$$\boldsymbol{u}''(x,t)=-\omega^2\boldsymbol{u}^1(x)\sin(\omega t)+\omega^2\boldsymbol{u}^2\cos(\omega t)$$

Substituting the expressions obtained for $u(x,t), u'(x,t), u''(x,t)$ into the initial statement of the equation and neglecting its initial conditions, we obtain a variational problem for the displacement amplitudes

$$\begin{cases} \textit{given vectors } l^1, l^2 \in L^2(V') \textit{ and circular frequency } \omega = const > 0 \\ \textit{find vectors } u^1, u^2 \in V := \{v \in H^1(\Omega),\ v=0 \textit{ on } \Gamma_U\} \textit{ such that} \\ -\omega m\left(u^1,v^1\right)-\omega c\left(u^2,v^1\right)+a\left(u^1,v^1\right)=l^1,v^1\ \forall v^1\in V \\ -\omega m\left(u^2,v^2\right)+\omega c\left(u^1,v^2\right)+a\left(u^2,v^2\right)=l^2,v^2\ \forall v^2\in V \end{cases}$$

where bilinear forms and linear functional have the following form.

$$\begin{cases} m(u,v)=\int_\Omega \rho u_j v_j dx \\ a(u,v)=\int_\Omega a_{ijkm} e_{ij}(u) e_{ij}(v) dx \end{cases}, \begin{cases} c(u,v)=\int_\Omega c_{ijkm} e_{ij}(u) e_{ij}(v) dx \\ l,v=\int_{\Gamma_\sigma} \hat{\sigma}_i v_i d\gamma \end{cases}$$

Energy balance equation

By combining the equations given above we can construct corresponding energy balance equation

$$\left[u_H^{12}+u_H^{22}\right]-\omega^2\left[u_V^{12}+u_V^{22}\right]=<l^1,u^1>+<l^2,u^2>$$

$$\omega\left[\left|u^1\right|_V^2+\left|u^2\right|_V^2\right]=<l^2,u^1>-<l^1,u^2>$$

Galerkin discretization

Let us use classical Galerkin discretization scheme. So, we choose a finite-dimensional subspace $V_h \subset V$ s.t. $dim V_h = N = N(h) \to \infty, h \to 0$ and basic functions:

$$u_h^1(x)=\sum_{k=1}^{N} U_k^1 \varphi_k^1(x),\ u_h^2(x)=\sum_{k=1}^{N} U_k^2 \varphi_k^2(x), \forall x \in \Omega$$

Putting this function into the variational formulation, we receive the following system of equations in matrix form

$$\begin{bmatrix} \mathbf{A}-\omega^2\mathbf{M} & -\omega\mathbf{C} \\ \omega\mathbf{C} & \mathbf{A}-\omega^2\mathbf{M} \end{bmatrix} \begin{bmatrix} \mathbf{U}^1 \\ \mathbf{U}^2 \end{bmatrix} = \begin{bmatrix} \mathbf{L}^1 \\ \mathbf{L}^2 \end{bmatrix}$$

where the matrices M, A, C, and vector L are calculated by corresponding bilinear forms and linear functionals.

Numerical experiments

For our experiments, let us take two-dimensional case. Consider as a substance fuel oil with the following physical properties: $\rho=890\frac{kg}{m^3}$, $c=1360\frac{m}{s}, \eta=2.022\, Pa \bullet s$, which fills a rectangular area with a side of 0.002m. On the part of the right side of the rectangle marked in red in (*Fig. 1*), the boundary conditions are defined as follows: $\sigma^1=(1,0), \sigma^2=(0,0)$. On all the remaining parts of the boundary of the region zero displacements $u^1=u^2=(0,0)$ are imposed.

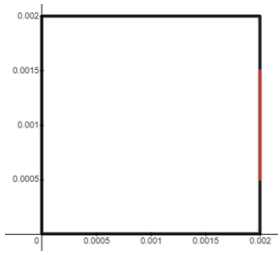


Figure 1. Rectangular area of experiment.

For the experiment, suppose a grid of size 75x75 and a frequency of 1 MHz. Results of computations are listed below.

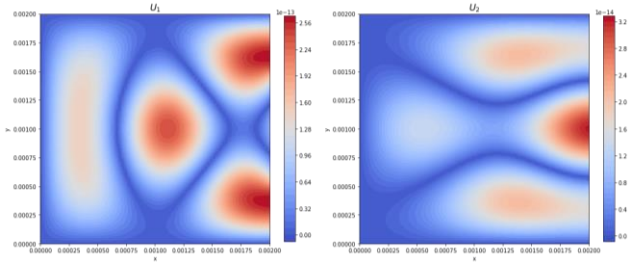


Figure 2. Displacement (mesh: 75x75, $\omega=10^6$).

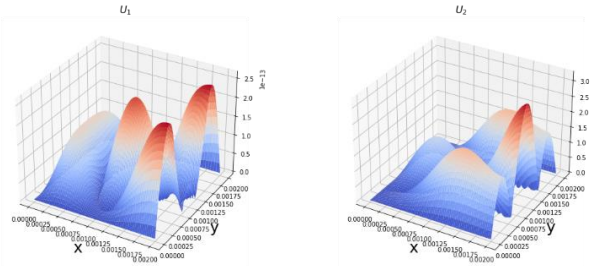


Figure 3. Displacement (mesh: 75x75, $\omega=10^6$).

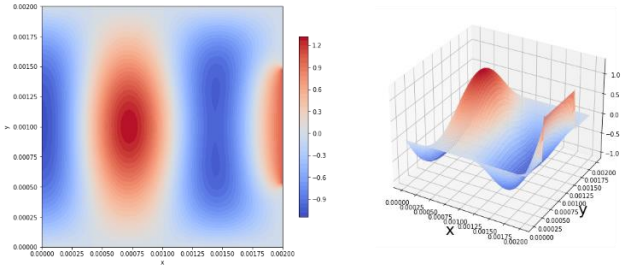


Figure 4. Pressure for U_1.

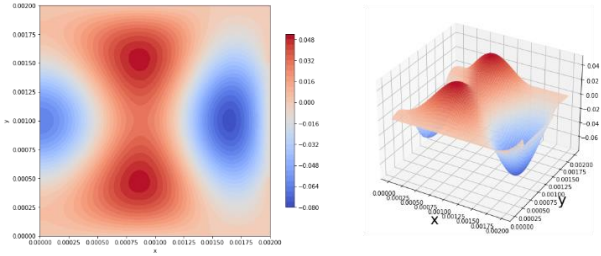


Figure 5. Pressure for U_2.

For this problem also performed a study of the convergence of numerical solutions

Table 1. Convergence at 0.5 MHz mesh densification

Mesh	$\|U_1\|_{H1}$	$\|U_2\|_{H1}$	$\|U_{1_k} - U_{1_{k-1}}\|_{H1}$	$\|U_{2_k} - U_{2_{k-1}}\|_{H1}$	k_1	k_2
16x16	7.7878e-13	9.564063e-15	5.00e-15	1.73e-16	-	-
32x32	7.7890e-13	9.565149e-15	1.19e-16	1.08e-18	-	-
64x64	7.7891e-13	9.565186e-15	1.63e-17	3.63e-20	2.87	4.90
128x128	7.7892e-13	9.565187e-15	4.48e-18	1.13e-21	1.86	4.99

Considering the performed numerical examinations at different parameters of space and time discretization, we can confirm the convergence of the chosen methods and the reliability of the developed and used software.

References

1. Johnson C. Numerical Solutions of Partial Differential Equations By The Finite Element Method / C. Johnson. – Dover Publications, 2009. – 278 pp.
2. Horlatch V. Formulation and well-posedness of the variational problem of viscous heat-conducting fluid acoustics / V. Horlatch, I. Klymenko and G. Shynkarenko // Journal of Numerical and Applied Mathematics – 2012. № 3 (109) – pp. 53-71.
3. Horlatch V. Finite element analysis of forced steady-state thermoacoustic waves in dissipative fluid / V. Horlatch, I. Klymenko, H. Shynkarenko // Manufacturing Processes. Actual Problems – 2013. Vol.1. Basic Science Applications. Opole: Politechnika Opolska, 2013. pp. 47-66.
4. Milchanovskyi V. Numerical Modeling of Acoustic Wave Propagation in Thermohydroelastic Systems / V. Milchanovskyi // International Student Scientific Conference of Applied Mathematics and Computer Science, May 6-7, 2021. – Lviv. – pp. 11-15. Available from: https://ami.lnu.edu.ua/wp-content/uploads/2022/02/ISSCAMCS-2021.pdf
5. The FEniCSx computing platform. Available from: [https://fenicsproject.org/]
6. Logg A. Automated Solution of Differential Equations by the Finite Element Method. The FEniCS Book / A. Logg, K.-A. Mardal, G. Wells. – Springer, 2012. – 723 p.

РОЗРОБКА ЧАТ-БОТА МОВОЮ ПРОГРАМУВАННЯ PYTHON

Христина Павліковська

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

Khrystyna.Pavlikovska@lnu.edu.ua

Вступ

Сучасні інформаційні технології не стоять на одному місті, їх розвиток відбувається все стрімкіше та стрімкіше. Вже відомо багато прикладів цього: розумний дім, 3D друк, електромобілі та багато іншого. Це однозначно позитивно впливає на наше життя. Набирають популярності системи віртуального спілкування, зокрема, чат-боти, за якими бачать значні переваги у використанні.

Упродовж багатьох років досліджують питання застосування систем віртуального спілкування на основі штучного інтелекту. Проблема віртуального спілкування актуальна через швидкий доступ до інформації, можливість одночасної роботи в системі багатьох користувачів, обмін інформацією, взаємодію з метою розв'язання будь-яких питань, підтримку навчання, комунікації з клієнтами та партнерами по бізнесу, проведення аналітичних досліджень, збору необхідної інформації, підвищення кваліфікації та інших переваг.

Головними питаннями у створенні систем спілкування є розробка моделі спілкування, моделі учасника спілкування, розвиток засобів, насамперед семантичних та прагматичних, описи навколишнього середовища (моделі мови, моделі користувача, моделі навколишнього середовища, моделі системи спілкування). Тож для розв'язання цих питань необхідно визначення принципів роботи, особливостей імітації мовної поведінки людини у процесі спілкування, розробка моделі спілкування, написання чат-бота. Багато програм-співрозмовників створені з урахуванням штучного інтелекту [1].

При розробці таких програм необхідно знати психологію і, навіть, принципи побудови фраз людської мови. Якщо правильно визначити мовні обмеження та предметну область, то наявними методами можна отримати системи, придатні для спілкування. З погляду теорії мови та спілкування необхідна розробка семантичного опису структур текстів та речень. З точки зору моделі навколишнього середовища основним обмеженням є відсутність засобів для представлення.

Навіть, якщо ви не розмовляли з Siri або Алісою, швидше за все, ви стикалися з чат-ботом. Він часто з'являється у вигляді діалогового вікна з дружнім вітанням: «- Дякуємо, що відвідали наш сайт. Як я можу вам допомогти?»

Залежно від тематики сайту, ви можете запитати про зламаний прилад у вашому домі або порадитись з приводу інвестицій. Чат-бот запрограмований на відповіді і навіть здатний ставити додаткові запитання.

Від чат-ботів, які обслуговують клієнтів онлайн, до особистих помічників у наших будинках – вони проникають у наше життя та часто виявляються корисними. Але як вони працюють? Яким чином можуть стати нам у пригоді?

Головна ідея

Принцип роботи чат-бота полягає в реалізації етапів: бот приймає вхідні повідомлення, аналізує їх та відсилає результат виконання та/або виконує команду. Отже, спілкування в чат-ботах здійснюється шляхом введення повідомлень та виведення відповіді (думки) співрозмовника. Тут можливі два види розмови: звичайна розмова чи обговорення важливого питання. Але, на відміну від розмови людей, програма не має гнучкого інтелекту, тому більшість віртуальних співрозмовників запрограмовані на ведення нескладної бесіди [2].

Telegram – це безплатний месенджер, який допомагає спілкуватися з іншими користувачами, надсилати файли різних розмірів та форматів. Головна ідея ресурсу - вільне та безпечне спілкування. Під час використанням цього додатку, користувачі мають можливість: спілкуватися в мережі з користувачами всього світу, також за допомогою аудіовикликів; отримувати та надсилати файли різних форматів (фото, аудіо, відео тощо); також використовувати хмарне сховище даних [3].

Telegram забезпечує комфортне середовище для користувачів та чат-ботів. Спілкування з чат-ботом у додатку не може бути складнішим, ніж зі звичайними людьми. Їх також використовують велика кількість користувачів для виконання різних завдань.

У багатьох царинах використовують чат-ботів для покращення або оптимізації обслуговування клієнтів та електронної комерції:

- Чат-боти для обслуговування клієнтів: багато компаній використовують ці сервіси як перший контакт, коли клієнтам потрібна допомога. Практично в кожній галузі компанії використовують чат-ботів, щоб

допомагати клієнтам легко переміщатися по їх вебсайт, відповідати на прості запитання та направляти людей до відповідних контактів.

- Чат-боти в електронній комерції: роздрібні компанії та телекомунікаційні провайдери використовують чат-ботів як додатковий канал взаємодії зі своїми клієнтами. Бот необхідний для ведення клієнтів через лінійний потік процесів, щоб виконати запити чи транзакції [3].

- Віртуальні помічники в чаті: особисті помічники, такі як Siri, Cortana та Аліса, стали популярними, оскільки їх переваги доступні і вони легко впровадилися в повсякденне життя споживачів. Люди використовують їх для швидкого отримання інформації, планування графіка та взаємодії з функціями розумного будинку.

Чат-боти спілкуються за допомогою промови або текстових повідомлень. Обидва варіанти можливі завдяки технологіям штучного інтелекту, таким як машинне навчання та обробка природної мови.

Чат-бот – це форма розмовного штучного інтелекту, призначена для спрощення взаємодії людини з комп'ютером. Використовуючи чат-ботів, комп'ютери можуть розуміти і реагувати на прихід людини через усне чи письмове мовлення [4].

При створенні проєктів, головною частиною є підбір засобів для розробки. Мова програмування Python - об'єктно-орієнтована, високорівнева, інтерпретована; вона активно розвивається та підходить для розв’язання багатьох задач. Python вважається одною з найпопулярніших мов програмування та часто використовується при створенні різних додатків для телефонів і комп'ютерів, у машинному навчанні та в багатьох інших сферах [5].

Ця мова програмування має простий та логічний синтаксис, тому код можна досить легко прочитати і сприйняти користувачам.

Отже, за допомогою цієї мови я і реалізовувала свій проєкт. Чат-бот призначений для того, щоб допомагати користувачам, а призначенням мого помічника буде рекомендація книг. Коли не знаєте, що можна почитати після продуктивного дня, чат-бот допоможе у виборі твору. Перед цим запитає про улюблений жанр та про інші необхідні параметри.

Висновки

Чат-боти - один з перспективних сучасних засобів, що дозволяють за допомогою мови програмування, автоматизувати багато процесів. Вже досить багато компаній використовують це нововведення для зв’язку з

користувачами, збору інформації та продаж. Чат-боти дозволяють уникнути багато проблем, які спричиняють малу ефективність та погану якість працездатності.

Функції чат-бота:

- автоматизація обробки запитів;
- здійснення пошуку та збір даних;
- надсилання відповідей та підтвердження;
- взаємодія із зовнішніми системами;
- ініціація запитів до контрагентів.

Також варто зазначити, що завдяки месенджерам чат-боти набули розвитку. З'явилися різні види для різних цілей. Основне завдання чат-ботів - вести діалог з користувачем та відповідати на його запити, спираючись на свою базу знань. Попри свої завдання чат-боти створюються для ведення діалогу користувачем, тим самим імітуючи співрозмовника або фахівця.

Чат-боти можуть виконати різні завдання: консультувати клієнтів, допомагати знайти потрібний файл, повідомляти новини, замовляти їжу, бути співрозмовниками та все в рамках однієї програми, на платформі якої реалізовано бот. Це зменшує необхідність встановлювати безліч додатків, тому що багато необхідних користувачеві запити можуть виконуватися в рамках однієї програми (месенджера).

Список використаної літератури

1. Проблеми віртуального спілкування - Леонтович О. А. // Полеміка. — 2000.
2. Особливості віртуального спілкування за допомогою чатів / Г. Д. Ушакова, Ю. В. Балабанова // Філологічний журнал: міжвишівська збірка наукових статей. - 2004. - С. 59 - 61.
3. Telegram. [Електронний ресурс]. - https://telegram.org/faq#q-what-is-telegram-what-do-i-do-here
4. [Електронний ресурс]. - https://www.techtarget.com/searchcustomerexperience/definition/chatbot
5. Python. [Електронний ресурс]. - https://www.python.org/about/

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ТЕСТУВАННЯ ПРОДУКТИВНОСТІ ПЕРСОНАЛЬНИХ КОМП'ЮТЕРІВ

Сергій Тороній

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

toroniyserhiy@gmail.com

З приходом комп'ютерної техніки життя людства кардинально змінилося. Дедалі більше розрахунків почали виконувати комп'ютери, що значно пришвидшило процеси технічного прогресу. На сьогоднішній день, з урахуванням росту продуктивності комп'ютерної техніки, значно збільшилась кількість та складність задач, які потрібно виконувати. Очевидно, що не у всіх є можливість отримувати найпередовіші новинки в технологічному просторі. Саме так і народжується проблема недостатньої продуктивності для поставлених задач перед комп'ютером. Тому актуальною темою є оцінка швидкодії персональних комп'ютерів. Для визначення таких оцінок були створені програми для тестування їх продуктивності при навантаженнях – бенчмарки. Деяка частина з нас, перед тим як придбати програмне забезпечення, звертала увагу на мінімальні та рекомендовані системні вимоги, у яких були подані назви комплектуючих комп'ютера. Ці дані були отримані саме в результаті тестування бенчмарками. Майже кожен з тих, хто пов'язаний з сферою інформаційних технологій – програміст, музикант, геймер, криптомайнер чи відеомонтажер, цікавиться про вимоги продукту з яким буде виконуватись подальша робота та систему, яка ці вимоги задовольняє. *ToroBench* – один з таких бенчмарків для тестування ПК.

Метою роботи є створення одного з таких продуктів для тестування продуктивності персональних комп'ютерів швидким та простим способом з реалізацією наступних можливостей:

- тестування швидкодії виконання арифметичних, тригонометричних стрічкових та інших операцій та алгоритмів для CPU
- тестування швидкодії виконання операцій пов'язаних з майнингом криптовалют та паралельними операціями для GPU
- тестування температурних показників на присутність троттлингу при тривалих та важких навантаженнях на ПК
- реалізація онлайн рейтингу ПК за продуктивністю, автооновлення застосунку та кастомізація програми

Даний продукт містить в собі онлайн рейтинг для порівняння своїх результатів з результатами будь-якого іншого користувача. Отже, основною ціллю є створення та розробка безкоштовного, швидкого бенчмарку з швидким та простим тестуванням продуктивності ПК на базі ОС Windows з використанням технологій та бібліотек:

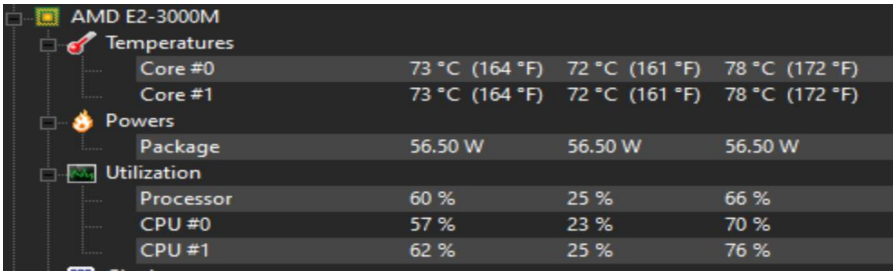
- Windows Forms C#
- OpenCL
- Dropbox API
- CUDAfy.NET
- Guna Framework
- AutoUpdater.NET

Torobench програмно реалізований на базі Windows Forms та .NET за допомогою мови програмування C#. Оскільки підтримка застосунку була націлена на операційну систему Windows, то було обрано саме такі основні технології. Дані рейтингу користувачів зберігаються в хмарному сховищі Dropbox, доступ до цих даних відбуваєтсья за допомогою Dropbox API. Для інтерфейсу програми було окремо використано Guna Framework через деякі обмеження створення дизайну стандартними інструментами Windows Forms. Оновлення застосунку реалізовано за допомогою відкритої бібліотеки AutoUpdater.NET. Вбудоване автооновлення дозволяє не шукати актуальні версії додатку в мережі Інтернет, що значно робить цю процедуру зручнішою для користувача, а також дозволяє підтримувати додаток в актуальному стані. Щоб зробити алгоритми для графічного процесора незалежними від розробника відеочіпів, їх було написано на низькорівневій мові програмування OpenCL та використана бібліотека для програмного доступу до графічного процесора - CUDAfy.NET.

З використанням вищеописаних технологій, застосунок тестує на продуктивність основні комплектуючі будь-якого ПК, будучи повністю незалежним від виробника цих комплектуючих. Основна інформація про ПК зображена на головному вікні програми. За основу тестування центрального та графічного процесора взято алгоритми пошуку простих чисел, арифметичні та тригонометричні операції, піднесення до степеня, визначення швидкості хешування та різні паралельні алгоритми, як от множення великих за розміром матриць. Крім того важкими для ЦП є операції з великою кількістю знаків після коми та операції із стрічковими даними, наприклад – конкатенація, які також використані при тестуваннях. Також реалізовано можливість запуску

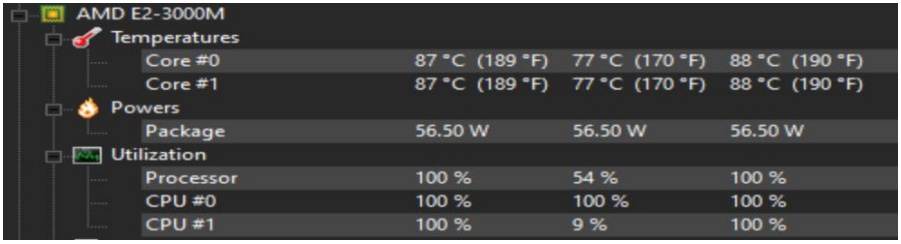
стрес-тестування ПК, з моніторингом температур, навантажень та показників троттлингу в реальному часі. Результати тестів користувачів після успішного тестування CPU і GPU автоматично додаються до рейтингу, який містить найкращі результати тестувань та дані про ПК з можливістю пошуку та сортування записів. Також доступна функція оновлення застосунку, в якій є можливість пропуску версії додатку або нагадування оновлень. До додаткових можливостей можна віднести кастомізацію вигляду додатку - вибір світлої, темної чи системної теми застосунку.

Порівнявши продукт з відомим аналогом – *Geekbench 5*, можна зробити наступні висновки та порівняння. По-перше, більшість аналогів *ToroBench* не є повністю безкоштовними, зокрема й *Geekbench 5*. По-друге, *Geekbench 5* не містить функціоналу стрес-тестування CPU/GPU, вбудованого рейтингу та вимірювання швидкості майнингу криптовалют. Навантаження під час тестувань та показники температур комплектуючих є меншими (див. Рис. 1) за навантаження та температури під час тестувань застосунком *Torobench* (див. Рис. 2), що є ключовим в програмному забезпеченні такого типу. Дані температур та навантажень отримано тестуванням за допомогою застосунку CPUID HWMonitor, який показує детальнішу інформацію порівняно із вбудованим в ОС Windows Диспетчером завдань.



AMD E2-3000M			
Temperatures			
Core #0	73 °C (164 °F)	72 °C (161 °F)	78 °C (172 °F)
Core #1	73 °C (164 °F)	72 °C (161 °F)	78 °C (172 °F)
Powers			
Package	56.50 W	56.50 W	56.50 W
Utilization			
Processor	60 %	25 %	66 %
CPU #0	57 %	23 %	70 %
CPU #1	62 %	25 %	76 %

Рис.1 Показники температур та навантаження CPU при тестуванні Geekbench 5



AMD E2-3000M			
Temperatures			
Core #0	87 °C (189 °F)	77 °C (170 °F)	88 °C (190 °F)
Core #1	87 °C (189 °F)	77 °C (170 °F)	88 °C (190 °F)
Powers			
Package	56.50 W	56.50 W	56.50 W
Utilization			
Processor	100 %	54 %	100 %
CPU #0	100 %	100 %	100 %
CPU #1	100 %	9 %	100 %

Рис.2 Показники температур та навантаження CPU при тестуванні Torobench

Порівнюючи з також не менш відомими бенчмарками *CPU-Z* та *GPU-Z* можна дійти висновку, що в *CPU-Z* та *GPU-Z* не вбудовані додаткові функції (онлайн рейтинг та вимірювання швидкості хешування GPU) та й взагалі ці програми поділені між собою за функціоналом, що зрозуміло за назвами застосунків.

Висновки

Створено десктопний застосунок для різного виду тестування продуктивності персональних комп'ютерів та іншими новими додатковими функціями для бенчмарків. Працюючи над даним додатком було закріплено здобуті знання в галузі десктопної розробки, роботи з хмарними сховищами та використання інших суттєвих бібліотек для застосунків такого типу – бенчмарків. В ході роботи було здобуто нові знання у створенні дизайну застосунку. Глибше досліджено роботу комплектуючих ПК та програмна взаємодія з ними.

Список літератури

1. OpenCL Overview. [Електронний ресурс]. – Режим доступу: https://www.khronos.org/opencl/
2. Windows Forms documentation. [Електронний ресурс]. – Режим доступу: https://docs.microsoft.com/en-us/dotnet/desktop/winforms/?view=netdesktop-5.0
3. Developers - Dropbox. [Електронний ресурс]. – Режим доступу: https://www.dropbox.com/developers
4. How expensive is an operation on a CPU? - StreamHPC. [Електронний ресурс]. – Режим доступу: https://streamhpc.com/blog/2012-07-16/how-expensive-is-an-operation-on-a-cpu/
5. Guna Framework. [Електронний ресурс]. – Режим доступу: https://gunaui.com/
6. .NET Documentation. [Електронний ресурс]. – Режим доступу: https://docs.microsoft.com/en-us/dotnet/?view=net-5.0
7. CUDAfy.NET. [Електронний ресурс]. – Режим доступу: https://github.com/lepoco/CUDAfy.NET

ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ ЗАРЯДНОЇ СМАРТ-СТАНЦІЇ AKER

Олег Іванків, Андрій Веліховський

Дублянський опорний ліцей імені Героя України Анатолія Жаловаги;
Львівський національний університет імені Івана Франка

oleguk0509@gmail.com, andriygh2326@gmail.com

Вступ

Однією з основних тенденцій розвитку інформаційних технологій є щорічне стрімке збільшення використання у повсякденному житті мобільних пристроїв. Основна маса таких пристроїв мають вбудоване джерело живлення (акумуляторну батарею), яке потребує зарядки. З огляду на сказане, виникає потреба у зростанні кількості ефективних портативних зарядних пристроїв.

На ринку зараз доступні зарядні станції, які мають ряд недоліків, основним з яких є обмеження кількості – не більше трьох, відсутність функцій швидкої зарядки і можливості контролю параметрів зарядки, обмеження загальної потужності до 30 Вт. Варто також додати, що в існуючих системах доволі часто відсутня система захисту від перегріву, перезаряду чи перерозряду акумуляторів, що значно знижує безпеку їхньої експлуатації.

Подолання цих недоліків зазвичай веде до значного росту ціни такого пристрою. Тому, на основі аналізу цих та деяких інших недоліків, запропоновано та реалізовано систему Aker, що відповідає наступним критеріям:

- безпека та зручність експлуатації;
- мультифункціональність;
- доступна ціна;
- швидкість зарядки та адаптивність.

Концепція та розробка пристрою

Aker – це портативна зарядна смарт-станція. Дана розробка має функцію швидкої зарядки та кольоровий tft дисплей, що дозволяє в зручному режимі відображати всі параметри зарядки. Також даний пристрій підтримує функцію безпровідної зарядки та має знімний акумулятор на 25000 mAh. Використовуючи термістори (для контролю температури акумуляторів і “battery management system”) вдалось значно збільшити безпеку експлуатації та транспортування.

Розробка пристрою розпочалась зі створення загальної концепції, та проведення необхідних розрахунків. Після чого в симуляторі easy eda

розроблено електронну схему та друковані плати. Наступним етапом стало створення макету пристрою та розробка програмного забезпечення. Паралельно з цим розроблялась 3D модель корпусу та елементів кріплення електроніки. Крім апаратної розробки проводився аналіз ринку та створення бізнес-моделі. Розроблений пристрій може бути використаний з метою оптимізації та розширення систем живлення пристроїв.

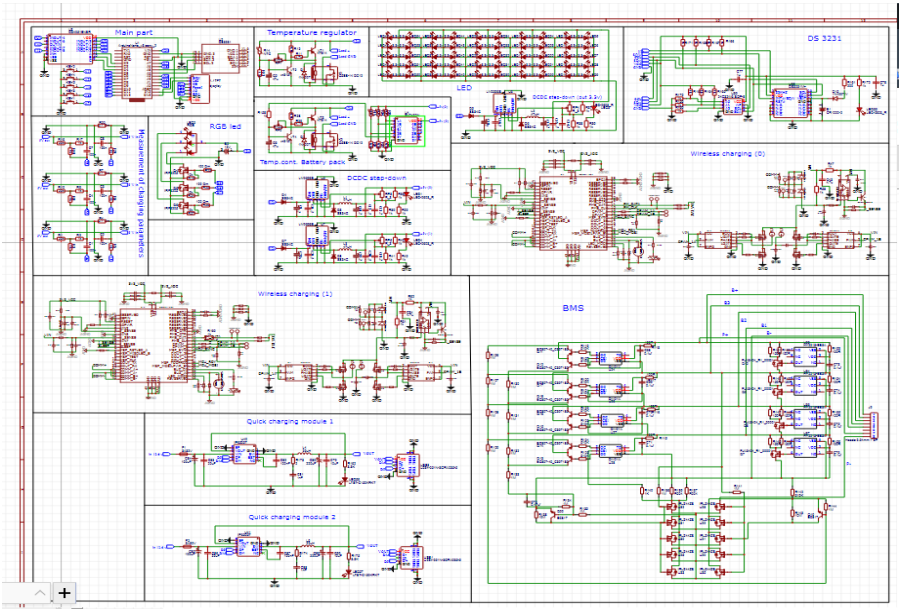


Рис. 1 Принципова схема Aker в симуляторі EasyEDA

Рис. 2. Тривимірна модель пристрою (зліва); зібраний пристрій (справа)

Висновки

В ході роботи було проаналізовано ринок зарядних станцій, проаналізовано велику кількість технічної документації на різнотипні електронні компоненти, проведено математичні розрахунки, та розроблено електронну схему пристрою. Також в даній роботі було написано програмний код для мікроконтролера. В функції даного коду входить: забезпечення функціонування 1.8 дюймового TFT SPI дисплея (написано інтерфейс, який складається з трьох основних панелей: панель часу, панель портів та панель налаштувань.) та системи контролю пристрою, LED та RGB підсвітки,

контроль параметрів зарядки та зчитування даних з мультиплексора CD4051. Створено та надруковано загальну 3D модель пристрою (див. додаток А), яку умовно можна розділити на 3 блоки: Основна частина; LED матриця; Акумуляторний відділ.

Список літератури

1. DC/DC-преобразователи: принципы работы и уникальные решения Maxim Integrated [Електронний ресурс] : URL:https://www.compel.ru/lib/134297
2. Quick Charge [Електронний ресурс] // Wikipedia / URL: https://en.wikipedia.org/wiki/Quick_Charge#:~:text=Quick%20Charge%20is%20a%20proprietary,while%20still%20maintaining%20compatibility%20to
3. How fast charging really works: your guide to wireless and wired fast charging [Електронний ресурс]: URL: https://www.androidauthority.com/fast-charging-explained-2-889780/
4. M. Rahman, A Review on Semiconductors Including Applications and Temperature Effects in Semiconductors // American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS), -2014, C. 50-70.
5. Гершунский Б. С., Романовская А. В., Ващенко Н. М. и др. Справочник по основам электронной техники. К.. Изд-во Киев. ун-та, 1974.
6. Дзюбин И. И. Тиристоры и электрических схемах. М., Энергия», 1972.
7. LM2596 Datasheet [Електронний ресурс] // Texas Instruments / URL: https://www.ti.com/product/LM2596
8. CD4051BNSR Datasheet [Електронний ресурс] // Texas Instruments / URL: https://pdf1.alldatasheet.com/datasheet-pdf/view/353843/TI/CD4051BNSR.html
9. TPS28225 Datasheet [Електронний ресурс] // Texas Instruments / URL: https://www.ti.com/lit/gpn/tps28225
10. BQ500211 Datasheet [Електронний ресурс] // Texas Instruments / URL: https://www.ti.com/lit/pdf/slvu536
11. HY2213 Datasheet // Hycon Technology 紘康科技 / URL: https://datasheet.lcsc.com/szlcsc/HYCON-Tech-HY2213-BB3A_C113632.pdf

OBJECT DETECTION AND SIZE RECOGNITION

Iryna Onyshkevych
Ivan Franko National University of Lviv
Faculty of Applied Mathematics and Informatics
onyshkevych.iryna@gmail.com

In this work, we discuss image processing techniques used for classical applications in the image detection field. We focus on the main problems: corners, edges, contours and size detection. The detection of size is believed to be the most challenging one since it typically utilizes the data obtained from previously detected corners, edges and contours of the object. Hence, it is of great importance to first discuss different methods for corner, edge and contour detection.

We start with a formal definition of the above-mentioned subproblems and an introduction of commonly used notations. All these problems are very well studied in the computer vision literature, and there are plenty of different methods and techniques for solving each of them.

The contour of the object plays an important role in image classification and object recognition and is closely related to previously discussed concepts, i.e., edge and corner. The goal is to determine the contour, which is the outline or a curve bounding the object. It can become a difficult task, especially when the contour is incomplete or unclosed as mentioned in [1]. The most common group of methods are pixel-based approaches [2]. There, the features are constructed and then employed to determine whether each pixel of the image belongs to a contour. To extract the discontinuity features, linear operators [3] (e.g., Sobel, Prewitt and Canny [4]) or quadratic filters [5] are introduced.

Another typical set of methods is edge-based approaches, which are aiming to determine whether the pre-detected curves or edges are contained within a certain contour. In many of these approaches, one must solve a global optimization problem, meaning that the information from the entire image is taken into the consideration simultaneously as explained in [6].

Here we present the results of contour detection obtained using the pixel-based method [7] from the OpenCV [8] library.

Fig. 1 a) group of small objects with the contrasting background, b) objects on the textured background of similar color but without shadows c) bright-colored objects with shadows on the contrasting background d) several objects of the color similar to the background with the prominent shadows

We observe that the method performs well even for small objects if their color differs from the background color (also when these two colors are not very contrasting), see Fig. 1 a, b. However, in case of similar colors or shadows, the solver struggles to determine the contour correctly (Fig. 1 c, d).

Finally, we present our results for size recognition. On Fig. 2 we have chosen Ukrainian 50 kopiyka, 2018, as a reference object. Its original diameter is 2,3 cm and we choose it to be represented by 150 pixels. For detection, we used the box of matches (original size 5,5 cm × 4 cm), a candle (diameter - 4 cm), a pen (16,8 cm× 0.6 cm) and an apple.

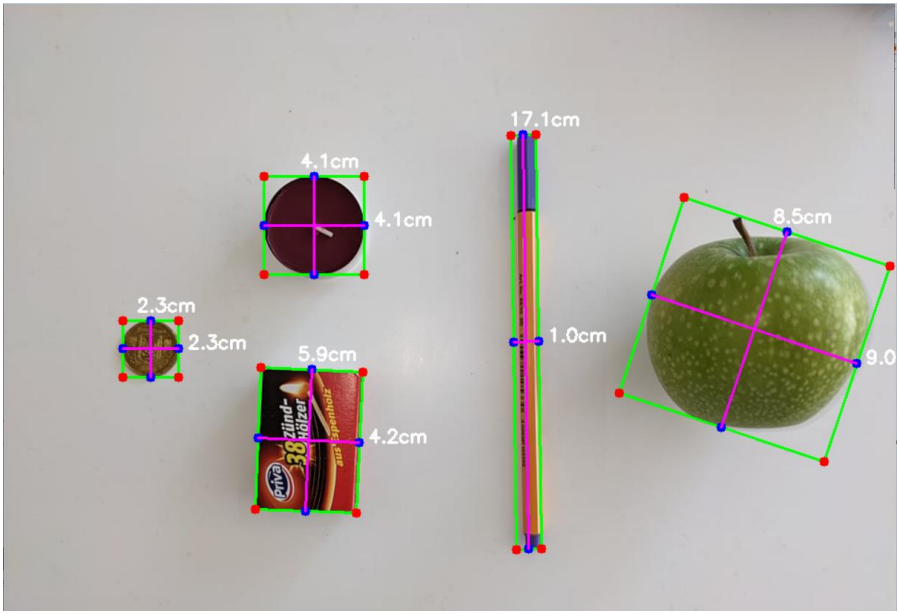


Fig. 2 Object size detection

In summary, image recognition, in particular, object contour detection and size recognition are of great theoretical and practical significance. In the present work, we started with the literature overview and comparison of different approaches to the problem. We discussed the results of contour detection obtained by different methods for different initial data. Finally, combining the methodology from discussed edge, corner and contour detection, we solved the problem of object size determination. We concluded by presenting the examples of size detection for some everyday objects and discussed the advantages and drawbacks of the used methods.

References

1. Gong, Xin-Yi & Su, Hu & Xu, De & Zhang, Zhengtao & Shen, Fei & Yang, Hua-Bin. An Overview of Contour Detection Approaches. International Journal of Automation and Computing, 2018

2. G. Papari, N. Petkov. Edge and line-oriented contour detection: State of the art. Image and Vision Computing, Vol. 29, no. 2–3, pp. 79–103, 2011
3. M. S. Nixon, A. S. Aguado. Feature Extraction & Image Processing for Computer Vision, 3rd ed., Amsterdam, Holland: Elsevier, 2012
4. J. Canny. A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8, no. 6, pp. 679–698, 1986
5. P. Perona, J. Malik. Detecting and localizing edges composed of steps, peaks and roofs. In Proceedings of the 3rdInternational Conference on Computer Vision, IEEE, Osaka, Japan, pp. 52–57, 1990
6. S. Wang, T. Kubota, J. M. Siskind, J. Wang. Salient closed boundary extraction with ratio contour. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 4, pp. 546–561, 2008
7. Satoshi Suzuki and others. Topological structural analysis of digitized binary images by border following. Computer Vision, Graphics, and Image Processing, 30(1):32–46, 1985\
8. Bradski G, Kaehler A. Learning OpenCV: Computer vision with the OpenCV library. " O'Reilly Media, Inc." 2008

ВИЯВЛЕННЯ ТА РОЗПІЗНАВАННЯ ОБ'ЄКТІВ ЗА ДОПОМОГОЮ АЛГОРИТМІВ МАШИННОГО НАВЧАННЯ ОНЛАЙН

Даніель Брецко

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

danielbrecko@gmail.com

На сьогоднішній день алгоритми знаходження та розпізнавання об'єктів є однією з найактуальніших тем машинного навчання, адже завдяки цьому створюються самокеровані автомобілі та робототехніка, відбуваються наукові відкриття у сфері медицини, покращуються системи безпеки.

Враховуючи релевантність даної теми, можна виділити основні завдання: проаналізувати поняття нейронної мережі, глибинного навчання та принципу backpropagation, дослідити структуру CNN та створити модель для виявлення та розпізнавання об'єктів, використовуючи глибоку згорткову нейромережу, натреновану на наборі даних COCO.

Для ефективного виконання поставлених завдань було використано такі моделі як: звичайні нейромережі, згорткові нейромережі та їхні вдосконалені версії (Mobile net v2), кожна з яких покращує різні аспекти точності, швидкодії та надійності результатів. А також інструменти для імплементації, які є невід'ємною частиною при роботі з нейромережами: Numpy, TensorFlow та OpenCV. Для досягнення основної цілі була використана саме бібліотека OpenCV, яка реалізовує алгоритми комп'ютерного зору та обробки зображень з відкритим кодом.

Однією з особливостей роботи із зображеннями є велика розмірність вхідних даних, оскільки кожен екземпляр є трьохвимірним масивом даних. При розпізнаванні зображень за допомогою повнозв'язних мереж може виникнути проблема перетренування. Щоб уникнути цього була створена згорткова нейронна мережа (Convolutional neural network, CNN), що складається з шарів входу та виходу, а також із декількох прихованих шарів, які зазвичай складаються зі згорткових (convolutional), агрегувальних (pooling) та повноз'єднаних (FC) шарів. На рис. 1 показана архітектура мережі.

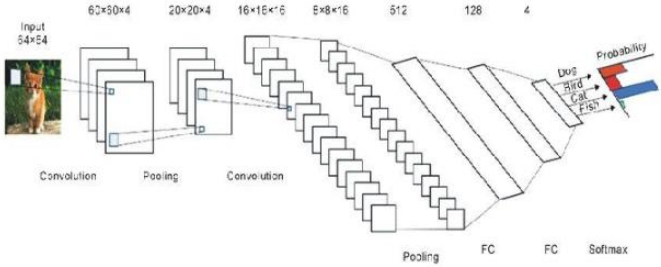


Рис. 1. Архітектура згорткової нейронної мережі [4]

За допомогою даної мережі можна провести класифікацію та локалізацію зображень. При класифікації зображень ми можемо ввести зображення в згорткову нейромережу з декількома шарами, яка виводить вектор значень, які подаються до функції активації softmax, яка виводить прогнозований клас. А для того, щоб локалізувати предмет, потрібно змінити нашу нейронну мережу, щоб мати ще кілька вихідних одиниць, які виводять обмежувальну рамку (Рис. 2)

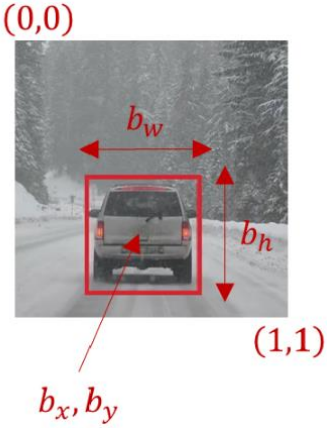


Рис. 2. Вигляд обмежувальної рамки та її параметри

Після того, як ми навчили цю мережу, зможемо використовувати її для виявлення машин за допомогою ковзних вікон. Ми робимо це таким чином, що якщо у нас є тестове зображення, як наведено нижче, ми починаємо з вибору розміру вікна, а потім вводимо в мережу тільки цю маленьку прямокутну область. Після розпізнавання цього вікна, ми посуваємо його на задану кількість кроків, після чого нейромережа знову розпізнає цю область, і

так до кінця зображення. Після проходження цілого зображення, можна змінити розміри вікна, щоб отримати точніші результати (Рис. 3).

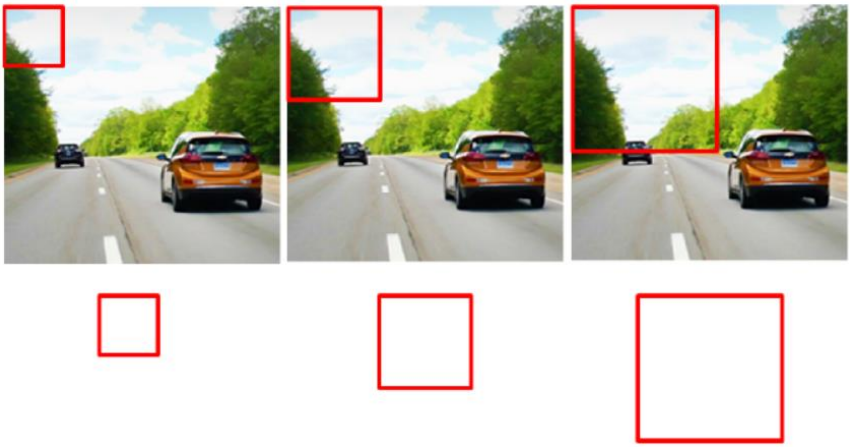


Рис. 3. Розпізнавання об’єктів з різни розмірами ковзних вікон

Загалом для реалізації згорткової нейромережі для виявлення та розпізнавання об’єктів онлайн, я використовував мову програмування Python з бібліотекою OpenCV та даними COCO2020, застосувавши архітектуру SSD Mobilenet v2.

У результаті цієї роботи вдалося створити програмний застосунок на основі нейромережі, яка здатна виявляти та розпізнавати предмети з оптимальною швидкістю та точністю для роботи онлайн (Рис. 3).

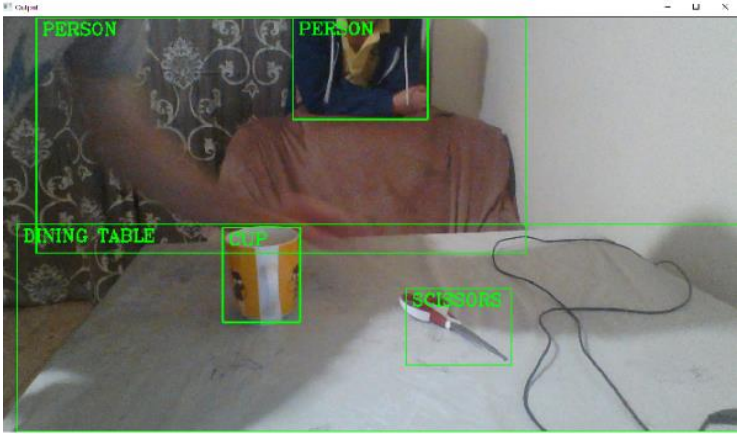


Рис. 3 Демонстрація роботи нейромережі

Помітно, що навіть при поганому освітленні, не найкращій якості зображення камери та випадках, коли предмети знаходяться «один в одному» (тобто один предмет знаходиться в обмежувальній рамці іншого), реалізована нейромережа справляється зі своєю роботою та розпізнає предмети з хорошою точністю та швидкістю.

Тому здобуті результати можна використовувати в цілях, де потрібне швидке та ефективне виявлення об’єктів.

Список літератури

1. MobileNetV2: Inverted Residuals and Linear Bottlenecks, https://arxiv.org/pdf/1801.04381.pdf
2. A Literature Study of Deep learning and its application in Digital Image Processing, https://www.researchgate.net/publication/317346042_A_Literature_Study_of_Deep_learning_and_its_application_in_Digital_Image_Processing
3. How Do Convolutional Layers Work in Deep Learning Neural Networks?, https://machinelearningmastery.com/convolutional-layers-for-deep-learning-neural-networks/
4. An Ensemble of Convolutional Neural Networks Using Wavelets for Image Classification, https://www.researchgate.net/publication/322957424_An_Ensemble_of_Convolutional_Neural_Networks_Using_Wavelets_for_Image_Classification
5. CNN Object Localization, http://datahacker.rs/deep-learning-object-localization/
6. COCO (Microsoft Common Objects in Context), https://paperswithcode.com/dataset/coco

ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПРОГНОЗУВАННЯ ДИНАМІКИ КУРСІВ КРИПТОВАЛЮТ ІЗ ВИКОРИСТАННЯМ ФРАКТАЛЬНОГО АНАЛІЗУ

Михайло Бордун

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

bordun.mykhailo24@gmail.com

Вступ

В останні декілька років криптовалюти набувають все більшої популярності, особливо це стосується Bitcoin, який є ключовим рушієм для всього крипторинку. Криптовалюта є більш безпечною альтернативою фізичній валюті, яку ми зараз використовуємо. Це все було здобуто завдяки шифруванню передачі конфіденційних даних за допомогою криптографічних протоколів, які є надзвичайно складними кодовими системами.

Актуальність теми

Об’єктом дослідження виступає динаміка курсів криптовалют, яку необхідно визначити з попередніх історичних даних. Ці дані представлені у вигляді часових рядів, тобто впорядкованої послідовності точок даних, розподілених на певний період часу. Аналіз часового ряду відноситься до визначення загальних закономірностей, які відображаються даними за певний період часу. Серед найпопулярніших методів моделювання та прогнозування часового ряду є авторегресійні моделі, найяскравішим представником серед яких виступає ARIMA (AutoRegressive Integrated Moving Average). Така класична модель як не дивно показує кращі результати прогнозування в порівнянні з моделями глибинного навчання для фінансових даних, оскільки вони є досить нестаціонарні, тобто хаотичні [1]. Узагальненням цієї моделі є фрактальна модель ARFIMA (AutoRegressive Fractionally Integrated Moving Average) [2], яка дає можливість моделювання часових рядів з довгою пам’яттю [3]. Загалом такого виду узагальнення дозволяє здійснювати необхідний аналіз різних часових рядів з врахуванням тривалого шоку в часовому ряді. Хоч ARIMA є надзвичайно популярним методом прогнозування часового ряду, однак ARFIMA поки тільки користується популярністю в науковій спільноті. Проте варто сказати, що зацікавленість до фрактального аналізу зростає, і це вже питання часу, коли в таких популярних

мовах програмування як Python з'являться готові рішення для такого роду аналізу, що однозначно знизить вхідний поріг і збільшить його популярність.

Мета

З огляду на перспективність цього ринку, все більше людей вбачають в ньому потужний інвестиційний інструмент. Проте ринок криптовалют досить складний і не має конкретних регуляційних важелів. Тому метою моєї роботи є розроблення інформаційної та інтелектуальної системи, достатньо надійної для того, щоб допомогти інвестору у прийнятті рішення, вказавши ймовірний напрямок розвитку обраної ним криптовалюти.

Використані технології

Дослідження, пов'язані з аналізом часових рядів та побудови фрактального диференціювання, були проведені за допомогою мови програмування Python версії 3.6.5 з використанням бібліотек pandas версії 1.1.3 та numpy версії 1.19.2. Для прогнозування часових рядів була використана мова програмування R версії 4.1.3, разом з бібліотеками forecast версії 8.16 та arfima версії 1.8.0.

Проведені дослідження

В ході роботи було розглянуто історичні криптовалютні дані для п'яти найпопулярніших криптовалют. З огляду на часові ряди, можна побачити, що до 2017 року криптовалюти не користувалися широкою зацікавленістю, і відносно сьогоднішнього стану речей ріст ринку був дуже повільний, тому ці дані не використовувалися для тренування. Також, визначено, що показник кореляція Пірсона між Bitcoin та іншими розглянутими криптовалютами є досить високий з найменшим значенням 0.65, що свідчить про досить сильний взаємозв'язок частин цього ринку.

Для аналізу та прогнозування був використаний часовий ряд найпопулянішої криптовалюти Bitcoin. Було визначено завдяки декомпозиції сезонність та наявність тренду у визначеному ряді. Також, завдяки статистичним тестам стаціонарності ADF та KPSS, визначено, що часовий ряд нестаціонарний. А завдяки графіку автокореляції визначено наявність довгої пам'яті, що також підтверджує наявність показника Херста , визначеного R/S аналізом [4], рівному 0.93. Цей показник також показує наявність тренду.

Моделювання фрактальної моделі ARFIMA(p,d,q) було виконано як з автоматичним підбором параметрів, так і з ручним. Ручний підбір параметрів виконувався з оцінок довжини масиву параметрів Autoregressive (AR) моделі порядку p з графіку автокореляції та Moving Average (MA) моделі порядку q з

графіку часткової автокореляції побудованих з фрактально диференційованого ряду, диференціювання якого реалізованого методом фіксованого вікна [5]. А після визначення порядку AR (p) та MA(q) визначався показник d інтегрованості ряду. Вибір оптимальної моделі визначався по оцінці RMSE з огляду на різницю між тестовими даними.

Аналіз результатів

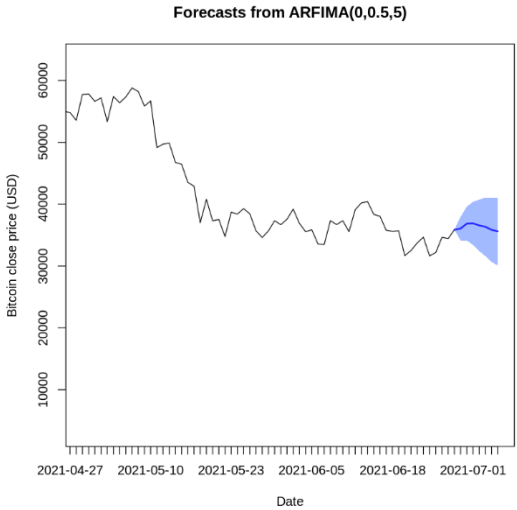


Рис. 1.Тижневе прогнозування Bitcoin моделлю ARFIMA(0, 0.4994476, 5)

Табл. 1.Точність прогнозування моделей з 1344 тренувальних, 30 тестових даних

Модель	RMSE	MAPE (%)
(auto) ARFIMA	2109.5	4.81
(auto) ARIMA	2317.6	5.35
(manually fitted) ARFIMA	2140.78	4.53

Табл. 2.Точність прогнозування моделей з 1367 тренувальних, 7 тестових даних

Модель	RMSE	MAPE (%)
(auto) ARFIMA	2136	5.76
(auto) ARIMA	1636.34	4.43
(manually fitted) ARFIMA	654.14	1.71

Табл. 3.Точність прогнозування моделей з 500 тренувальних, 30 тестових даних

Модель	RMSE	MAPE (%)
(auto) ARFIMA	2741.5	6.92
(auto) ARIMA	2308.63	5.3
(manually fitted) ARFIMA	2205.05	4.67

Можна побачити по моделям ARFIMA та ARIMA з автоматичним підбором параметрів, що збільшення як тренувальної так і тестової вибірки більш виразно показує перевагу фрактальної моделі, оскільки більше враховуються ефекти довгої пам'яті для ефективного прогнозування. Результати також показують, що автоматичний підбір параметрів не завжди є оптимальним і можна покращувати похибку моделі завдяки перебору моделей ARFIMA з різними параметрами AR (p) та MA (q), на основі яких визначається параметр фрактального диференціювання (d).

Висновок. У даній роботі розглянуто основні принципи аналізу часових рядів, а також дослідження наявності довгої пам'яті в них. На прикладі криптовалюти Bitcoin результати виконання прогнозування на повному наборі тренувальних даних також підтверджують доцільність використання моделі ARFIMA в порівнянні з моделлю ARIMA з огляду на різноманітні оцінки тестової похибки такі як RMSE, MAPE. В подальшому планується створити хмарну інфраструктуру з використанням Telegram-бота на Python, який би давав результати прогнозування для даних з біржі в реальному часі.

Використана література

[1] Shah V. Forecasting Market Prices using DL with Data Augmentation and Meta-learning: ARIMA still wins! / V. Shah, G. Shroff – 2021. – Available from: https://arxiv.org/abs/2110.10233

[2] Liu K. An Evaluation of ARFIMA (Autoregressive Fractional Integral Moving Average) Programs / K. Liu, Y. Chen, X. Zhang // Axioms. – 2017.– Vol. 6.– P. 1-16.

[3] Перцовский О. Моделирование валютных рынков на основе процессов с длинной памятью: Препринт WP2/2004/03 — М.: ГУ ВШЭ. – 2003.– 52 с.

[4] Safitri D. Gold price modeling in Indonesia using ARFIMA method / D. Safitri, Mustafid, D. Ispriyanti, Sugito // IOP Conf. Series: Journal of Physics: Conf. Series 1217.– 2019.– P. 7-9.

[5] De Prado M. L. Advances in Financial Machine Learning / M. L. de Prado // John Wiley & Sons, Inc., Hoboken, New Jersey. – 2018.– P. 75-91.

МОДЕЛЬ МАШИННОГО НАВЧАННЯ ДЛЯ РОЗПІЗНАВАННЯ АНГЛІЙСЬКОГО ДАКТИЛЬНОГО АЛФАВІТУ

Олег Смаль

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

oleh.smal@lnu.edu.ua

У сучасному світі, який прагне до інклюзивності та рівності, тема допомоги людям з вадами слуху є як ніколи актуальною. Розвиток моделей подібних цій, може спростити роботу, чи навіть повністю прибрати потребу у людських перекладачах, при тому генеруючи результати швидше та якісніше ніж люди.

На відміну від багатьох моделей, спрямованих на розв'язання цієї задачі, замість того, щоб використовувати методи глибокого навчання, було вирішено застосувати аналітичний підхід.

Цей алгоритм опрацьовує зображення в наступні три послідовні етапи, що є чітко взаємно відокремлені:

- Визначення ключових точок (суглобів) руки на зображенні.
- Нормалізація і балансування отриманих результатів.
- Застосування власне моделі.

Перший крок бере за основу популярну модель Google MediaPipe Hands, яка на вхід приймає зображення кисті руки та на виході повертає масив 21 точки у 3-вимірному просторі, де кожна з точок відповідає одному з ключових суглобів. На цьому етапі також відбувається фільтрування невалідних зображень: якщо MediaPipe не спроможна розпізнати суглоби кисті на зображенні, зображення вважається неприйнятним. Важливо зазначити, що MediaPipe одним з аргументів бере на вхід інтервал довіреності. Змінюючи його, ми можемо контролювати толерантність результуючої моделі до похибок у вхідних зображеннях. Проте очевидним спостереженням буде те, що зі зростанням допустимої межі похибки, модель стає прогресивно легковажною, і межа між класами стає менш чіткою. Також помітимо, що в силу обмеженості у виборі тренувального набору даних, занадто низька толерантнісь до відхилень зображення призведе до «недогодованості» моделі, і в свою чергу її невалідності.

На другому кроці відбувається проміжне опрацювання отриманих даних та очистка їх для тренування моделі.

На третьому кроці застосовується сама модель. При розв'язуванні цієї задачі за основу було прийнято ансамблевий метод «Випадкового Лісу» (далі ВЛ), оскільки він добре підходить для задач мультикласового передбачення.

Цей метод полягає у наступному:

В ретроспективі необхідно оцінити якість моделі. У цій роботі прийматимуться до уваги наступні аналітичні параметри:

1. Влучність — частка релевантних зразків серед знайдених.
2. Повнота — частка загального числа позитивних зразків.
3. f1-міра — агрегація перших двох.

На відміну від багатьох сучасних вербальних мов, інформаційна щільність дактильного мовлення є досить низькою. Попри те, що технічно, кожна дактилема відображає одну літеру відповідного алфавіту, через фізичну природу дактильного спілкування, кількість інформації, яку виробляє мовець за одиницю часу, є меншою ніж у мовця відповідної вербальної мови.

Як наслідок, переклад дактильних мов є критично вразливим до помилок.

Можливими розв'язками цієї проблеми могли б бути:

- Частотний аналіз літер алфавіту.
- Аналіз послідовностей дактилем як морфологічних структур.
- Аналіз послідовностей морфологічних структур.
- Ланцюги Маркова для токенізованих структур.

Джерела

1. Bruce P. Practical Statistics for Data Scientists / P. Bruce, A. Bruce // O'Reilly Media, Inc.—2020.
2. Müller A. Introduction to Machine Learning with Python / A. Müller, S.
3. Guido // O'Reilly Media, Inc.—2016.
4. Bishop C. Pattern Recognition and Machine Learning / C. Bishop, M. Jordan, J. Kleinberg, B. Schölkopf // Springer Science+Business Media, LLC.—2006.
5. McKinney W. Python for Data Analysis / McKinney W. // O'Reilly Media, Inc.—2017.
6. Python Scikit-Learn Library Documentation — Режим доступу: https://scikitlearn.org/stable
7. Python Google MediaPipe Library Documentation — Режим доступу: https://google.github.io/mediapipe/
8. Python OpenCV Library Documentation — Режим доступу: https://docs.opencv.org/4.x/
9. Python Pandas Library Documentation — Режим доступу: https://pandas.pydata.org/pandas-docs/stable/
10. Python NumPy Library Documentation — Режим доступу: https://numpy.org
11. ASL Alphabet Dataset — Режим доступу: https://www.kaggle.com/datasets/grassknoted/asl-alphabet/download

ПОРІВНЯННЯ ЕФЕКТИВНОСТІ ВИКОРИСТАННЯ РЕЛЯЦІЙНОЇ МОДЕЛІ ДАНИХ ТА МЕРЕЖЕВОЇ МОДЕЛІ ДАНИХ У ВИГЛЯДІ ГРАФУ ДО ТІСНО ПОВ'ЯЗАНИХ ДАНИХ НА ПРИКЛАДІ СОЦІАЛЬНОЇ МЕРЕЖІ

Артем Волобуєв

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

artem.volobuiev@lnu.edu.ua

Вступ

Історично, перші СКБД (системи керування базами даних) були нереляційними. Мережева модель даних також з'явилась одною із перших. Тим не менш, висока складність і жорсткість схеми БД (бази даних) стали одними із головних причин, за яких поступово нереляційні СКБД, побудовані на основі мережевої моделі, поступилися місцем реляційним. Проте наприкінці першого десятиліття 21-го століття, водночас, з поступовим поверненням зацікавлення у сторону нереляційних СКБД, почав зростати інтерес й до БД, у яких мережева модель була реалізована у вигляді графу. Це пов'язано з тим, що реляційний підхід, хоч і задовольняє більшість потреб середнього користувача, погано підходить для зберігання даних, що є зв'язками. У сучасному світі проблеми обробки даних з подібною структурою виникають постійно, з регулярним зростанням кількості користувачів соціальних мереж, а також розвиненням їх можливостей.

Мета

Враховуючи зростаючу роль графових БД для зберігання даних з графовою структурою, було вирішено провести дослідження щодо їх продуктивності у порівнянні з реляційними БД.

Використані технології

Дослідження були проведені за допомогою мови програмування C#, з використанням платформи ASP.NET. Як графова БД була обрана Neo4j, як реляційна PostgreSQL. Для зв'язку з Neo4j у веб-аплікацій була обрана бібліотека Neo4j.Driver. Для зв'язку з PostgreSQL Entity Framework та бібліотека Npgsql.EntityFrameworkCore.PostgreSql.

Порівняння реляційної та графової БД на основі розробленого веб-застосунку соціальної мережі

Для практичного порівняння можливостей реляційних та графових БД була створена веб-аплікація у вигляді соціальної мережі. До можливостей аплікації входять: логін та реєстрація; перегляд публікацій користувачів, на яких підписаний даний, та реакції на них; пошук користувачів серед усіх та серед підписників; перегляд сторінок інших користувачів. На сторінці іншого користувача є можливість переглянути «відстань до користувача», по аналогії з соціальною мережею LinkedIn (рис. 1).

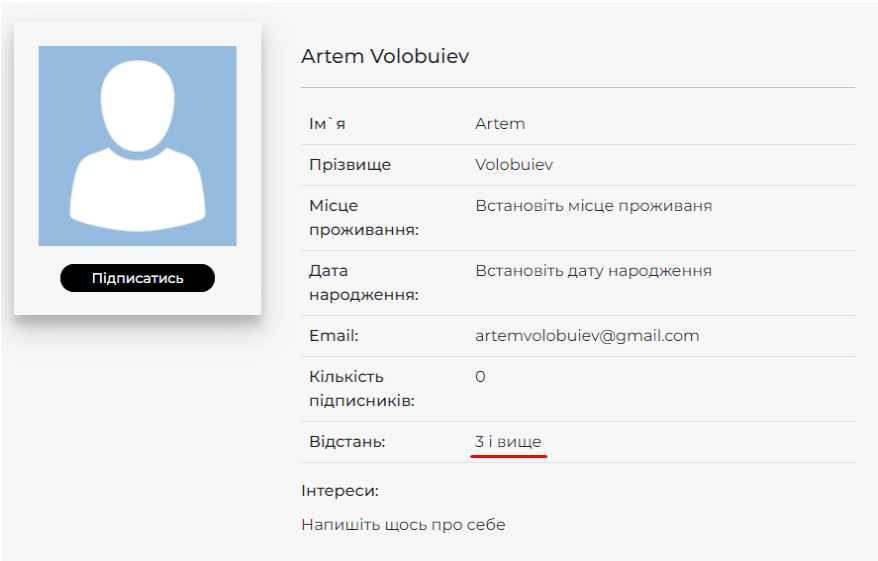


Рис. 1. Відстань до іншого користувача 1

Графова БД дозволяє знаходити найкоротший шлях до іншого користувача, а потім підраховувати його довжину. У реляційний БД для подібного доведеться або реалізовувати складну збережену процедуру, або проводити пошук найкоротшого шляху на стороні сервера. Очевидно, у даному випадку графова БД виграє як у швидкості виконання подібної процедури, так і у зручності.

Цікавою для порівняння також є можливість пошуку підписників для конкретного користувача. У цьому випадку можна здійснити пряме порівняння швидкості запиту на знаходження підписників у графовій та у

реляційній БД. Структура таблиці для зберігання підписок у PostgreSQL максимально проста (рис. 2).

Columns +

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
		Id	bigint			Yes	Yes
		SubscriberId	bigint			Yes	No
		SubscribeId	bigint			Yes	No

Рис. 2. Структура таблиці підписок

У таблиці є первинний ключ та два вторинних. Перший вказує на Id підписника, другий на Id людини, на яку перший підписаний. Id у даному випадку первинний ключ з таблиці користувачів. У графовій БД ж нема чіткої схеми і підписка – просто ребро між вузлами користувачів у графі.

Тест проводився на згенерованих 1080 записах і 1080 підписках між користувачами. Для порівняння було обрано одного користувача, якому навмисно було створено найбільше підписок – від 70 до 80. За результатами, в середньому запит до реляційної БД займав 55 мс., в той час як до графової БД лише 3 мс.

Висновки

У цій роботі для порівняння реляційної та мережевої моделі на основі графу була створена веб-аплікація у вигляді соціальної мережі, для якої структура даних про зв’язки між користувачами є природнім графом. Результати дозволяють стверджувати, що використання графових БД на таких даних сильно прискорює запити, а також дозволяє значно спростити розробку, оскільки графові БД надають вбудованих можливостей для роботи з графами.

Використана література

1. Troelsen A. W. Pro C# 9 with .NET 5: Foundational Principles and Practices in Programming 10th ed. Edition /A. W. Troelsen, P. Japikse. – Apress, 2021. – 1411 c.
2. Adam Freeman Pro ASP.NET Core 3 (Develop Cloud-Ready Web Applications Using MVC 3, Blazor, and Razor Pages) 8th ed. Edition. – Apress, 2018. – 1109 c.
3. Neo4j Cypher Manual [Електронний ресурс] – https://neo4j.com/docs/cypher-manual/current/

.NET-БІБЛІОТЕКА МАТЕМАТИЧНИХ ЗАСТОСУНКІВ APPLIEDMATHLIBRARY

Ростислав Байцар

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

Rostyslav.Baitsar@lnu.edu.ua

Вступ

У зв'язку з великою популярністю .NET платформи, сфера її застосування зростає, зростає також й потреба в готових програмних рішеннях, в тому числі – бібліотек.

Дана бібліотека є ідейним аналогом популярної Python бібліотеки "NumPy". Потреба в появі бібліотеки виникла з відсутності схожого рішення для мови C#.

У період здобування ступеня бакалавра, мені довелося виконувати багато лабораторних робіт з яких я "запозичав" цілі блоки коду. Вирішенням проблеми копіювання і є дана бібліотека.

Метою даної роботи є проектування та розробка .NET бібліотеки з широким набором функціоналу для розв'язування базових математичних задач, що у свою чергу вимагає забезпечення високої надійності методів та повну сумісність з мовами сімейства .NET.

Окрім цього, потрібно забезпечити просту структуру бібліотеки, документованість, низький поріг входження.

Модель поширення, контроль версій, ліцензія

Вихідний код є у вільному доступі у git за посиланням:

https://github.com/Rostik18/AppliedMathLibrary

Поширення відбувається за допомогою пакетного менеджера NuGet за посиланням:

https://www.nuget.org/packages/AppliedMathLibrary/

Бібліотека поширюється за ліцензію MIT.

Функціонал

У запропонованій бібліотеці, на даний час, реалізовано деякі основні засоби, що використовуються для математичних розрахунків, зокрема для лінійної алгебри, теорій прийняття рішень, математичної статистики тощо.

Об'єкти та функціонал, що вони реалізують:

- Matrix (методи: IsSquare, Rows, Columns, Indexer, Transpose, ToVectors, Subtract, Multiply, GetEnumerator);

- Vector (методи: Dimension, Indexer, CompareByValue, Comparable, Subtract, Sum, Divide, Multiply, Norm, DistanceBetween, GetEnumerator);
- Point (методи: Dimension, Indexer, GetEnumerator);

  Статичні класи та їх методи:
- GaussMethod для розв’язування СЛАР;

- ParetoMethods для пошуку кращих альтернатив за принципом Парето;
- SlaterMethods для пошуку кращих альтернатив за принципом Слейтера;
- Statistics (методи: Mean, Median, Mode, MathExpectation, Var, Std, GenerateRandomArray, RandomSampleByRule1, RandomSampleByRule2);

Висновки

У результаті розробки створено бібліотеку яку може використати кожен охочий у власних цілях.

Варто зауважити що це лише перша верша версія даного продукту. Кожен бажаючий може долучитися до покращення і розвитку даної бібліотеки, надіславши власні зауваження чи побажання.

Література

1. Василик О. Лекції з теорії і методів вибіркових обстежень. Навчальний посібник // О. І. Василик, Т. О. Яковенко / Київський національний університет імені Тараса Шевченка, ВПЦ "Київський університет", 2010. – 206 с.
2. Зеліско, В. Р. Основи лінійної алгебри і аналітичної геометрії // В. Р. Зеліско, Г. В. Зеліско / – Львів: Львівський національний університет імені Івана Франка, 2011. – 326 с/
3. Сеньо П.С. Теорія ймовірностей та математична статистика // П. С. Сеньо / К.: Центр навчальної літератури, 2004. – 448 с.
4. Шахно С.М. Практикум з чисельних методів // С.М. Шахно, А. Т. Дудикевич, С.М. Левицька / − Львів, 2013. − 133 с.
5. Install and manage NuGet packages in Visual Studio. [Електронний ресурс]. – Режим доступу: https://docs.microsoft.com/en-us/nuget/consume-packages/install-use-packages-visual-studio

МАТЕМАТИЧНА МОДЕЛЬ ПОШИРЕННЯ COVID-19

Аліса Трей

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

alisatrey88@gmail.com

Вступ

У цій роботі представлена математична модель SEIRD, яка враховує деякі особливості поширення COVID-19, а саме латентних носіїв вірусу . Зараз у світі існує велика кількість різних математичних моделей, що описують поширення епідемій. Проте, перебіг кожного інфекційного захворювання має свої особливості. І поява нових масових захворювань призводить до необхідності вносити корективи в наявні моделі.

Моделі, що базуються на диференціальних рівняннях, посідають особливе місце серед існуючих моделей поширення інфекційних захворювань. Можливо, вони менш реалістичні, ніж стохастичні моделі, але більш зручні для аналізу.

Використані технології

Дослідження були проведені за допомогою пакету Jupiter Notebook, Python.

Базова SEIRD-модель

Для опису епідемії COVID-19 з вищезгаданих моделей найбільш підходящою є SEIRD-модель[2], в якій передбачається, що захворювання має латентний період, тобто перші ознаки захворювання з'являються у зараженого через певний проміжок часу[1]. З урахуванням цього фактору населення ділиться на 5 груп:

- сприйнятливі ($S(t)$),
- латентні ($E(t)$),
- інфіковані ($I(t)$),
- одужавші ($R(t)$),
- померлі ($D(t)$).

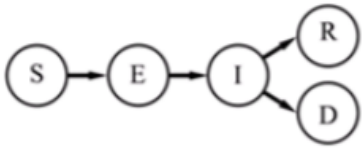


Рис. 1: Схема можливих переходів між групами SEIRD-моделі

Розглянемо чинники, що впливають динаміку виділених груп. Очевидно, що зараження та перехід у латентну фазу відбувається в результаті контакту сприйнятливих та інфікованих людей. Причому відносна частота такого контакту дорівнює $\frac{S(t)}{N} \cdot \frac{I(t)}{N}$. Отже, швидкість зміни частки латентних людей $\frac{d}{dt}\left(\frac{E(t)}{N}\right)$ містить пропорційний доданок $\frac{S(t)*I(t)}{N*N}$.

Зазначимо також, що людина, ставши латентною, не може стати такою повторно, тобто $\frac{d}{dt}\left(\frac{E(t)}{N}\right)$ містить доданок, пропорційний $\frac{S(t)}{N} \cdot \frac{I(t)}{N}$.

Отже,

$$\frac{d}{dt}\left(\frac{E(t)}{N}\right) = \beta\,\frac{S(t)*I(t)}{N*N}\,. - \delta\,\frac{E(t)}{N}.$$

Розмірковуючи аналогічним чином щодо інших груп популяції, отримаємо систему звичайних диференціальних рівнянь, що описують динаміку епідеміологічного процесу (SEIRD-модель):

$$\frac{dS(t)}{dt} = -\beta\,\frac{S(t)*I(t)}{N},$$

$$\frac{dE(t)}{dt} = \beta\,\frac{S(t)*I(t)}{N}, - \delta E(t),$$

$$\frac{dI(t)}{dt} = \delta E(t) - (\gamma + \mu)I(t),$$

$$\frac{dR(t)}{dt} = \gamma I(t),$$

$$\frac{dD(t)}{dt} = \mu I(t).$$

в якій:

β – коефіцієнт, який можна інтерпретувати як ймовірність отримання хвороби у разі контакту сприйнятливого індивідуума з інфікованим;

μ - коефіцієнт смертності;

γ – коефіцієнт, який можна інтерпретувати як швидкість одужання, тобто $\gamma = 1/T_1$, де T_1 – середній час хвороби;
δ – коефіцієнт, який можна інтерпретувати як швидкість переходу з латентної фази у фазу інфікування, тобто $\delta = 1/T_2$, де T_2 – середній час інкубаційного періоду.

Коефіцієнти для задачі підберемо, використовуючи дані розповсюдження COVID-19 у Китаї.

Згідно з наявними у відкритому доступі статистичними (усередненими) даними $T_1 = 18$, $T_2 = 5.2$, коефіцієнт смертності – 3.4% , число заражень від однієї інфікованої людини – репродуктивне число $R_0 = \frac{\beta}{\gamma} = 3.5$. Тоді $\gamma = 0.056$, $\delta = 0.192$, $\mu = 0,034$ (дане значення може коливатись в межах від 0 до 1), $\beta = 0.194$. [3][4]

Населення Китаю на 2019 рік становило 1,410 млрд. Час моделювання – 400 днів.

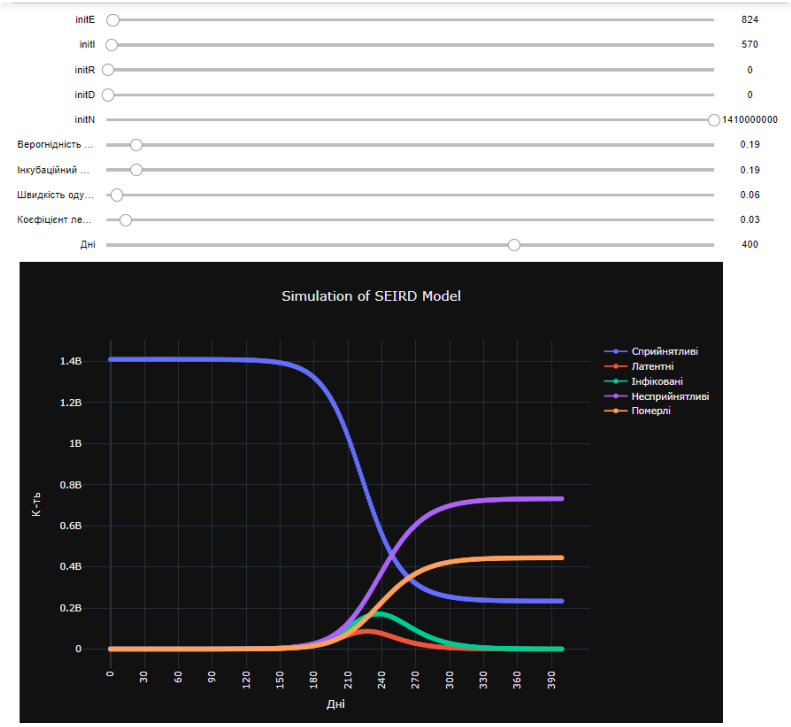


Рис.2: Результат моделі поширення SEIRD моделі

За допомогою програми, у якій реалізовано можливості змінювати параметри, ми отримали графік за заданих початкових умов.

Якщо спробувати змінити значення, наприклад вірогідність отримання хвороби з 0.19 на 0.89 та інкубаційний період з 0.19 та 1.56, то отримаємо наступний графік:

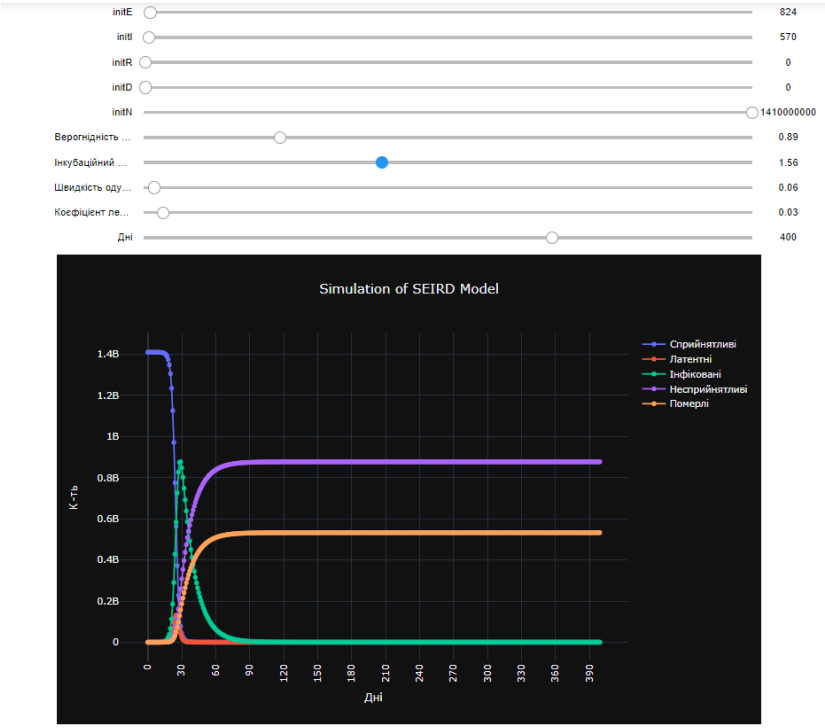


Рис. 3: Результат моделі поширення SEIRD моделі зі зміненими параметрами

Ми можемо побачити суттєву різницю та зрозуміти, що навіть незначна зміна параметрів може нести за собою великі зміни в прогнозуванні моделі хвороби. А саме: кількість хворих буде різко збільшуватись та охоплювати більшу кількість людей, збільшиться кількість померлих та несприйнятливих і відповідно зменшиться кількість сприйнятливих. Перебіг епідемії скоротиться

Висновок

У даній роботі були побудовані моделі з різними значеннями β та δ, які описуюсь розповсюдження коронавірусу. Було розглянуто, як змінюється

кількість інфікованих в залежності від початкових параметрів. Був зроблений прогноз кількості інфікованого населення і за певних параметрів він є подібний до реальних даних. Всі побудовані моделі є лише припущеннями, тому їх не варто сприймати як ідеальну модель, яка повністю відповідає дійсності. Звісно, що ця модель досить спрощена, і вона не враховує всі фактори, які впливають на розповсюдження. Наприклад, зменшення спроможності медичної допомоги через відсутність належної кількості тестів.

Література

[1] Едельштейн-Кешет Л. Математичні моделі в біології // Товариство промислової та прикладної математики, 2005. 586 с.

[2] Єрємєєва Н.І. Побудова модифікації SEIRD-моделі поширення епідемії, що має особливості COVID-19 // Вісник ТВГУ. Серія: Прикладна математика. 2020. № 4. С. 14-27

[3] О. І. Кріворотько С. І. Кабанихін Математичні моделі розповсюдження COVID-19 // НГУ

[4] Вхідні данні для моделі//
https://www.medrxiv.org/content/10.1101/2020.04.01.20049825v1.full.pdf

РЕАЛІЗАЦІЯ РІЗНОМАСШТАБНОГО МЕТОДУ СКІНЧЕННИХ ЕЛЕМЕНТІВ ДЛЯ ДВОВИМІРНОЇ ЛІНІЙНОЇ КРАЙОВОЇ ЗАДАЧІ

Дмитро Альянах

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

dmytro.alianakh@lnu.edu.ua

Вступ

Метод скінченних елементів (МСЕ) є одним із найпопулярніших методів для розв'язування диференціальних рівнянь у часткових похідних. Він широко застосовується в аеро-, морській, автомобільній та інших індустріях.

Тут розглядається модифікація методу скінченних елементів для випадку неоднорідного середовища під назвою різномасштабний метод скінченних елементів (РМСЕ) для певного типу двовимірних лінійних крайових задач, який запропонований у [1].

Постановка задачі

Розглянемо двовимірну крайову задачу

$$-div\big(k(x)\nabla p(x)\big) = f(x), x \in \Omega = (a,b) \times (c,d), \quad (1)$$

$$a,b,c,d \in \mathbb{R}, 0 < a < b, 0 < c < d, \qquad p = 0 \text{ на } \partial\Omega, \quad (2)$$

де $k(x)$— функція, яка характеризує неоднорідності, $p(x)$— шукана функція, $f \in L_2(\Omega)$ – задана права частина.

Домножимо (1) на $v \in V\left(V = \left\{p(x): p(x) \in W_2^{(1)}, p = 0 \text{ на } \partial\Omega\right\}\right)$ і врахуємо крайову умову, отримаємо варіаційне формулювання:

$$\int_\Omega k(x)\nabla p_h(x) \cdot \nabla v(x)dx = \int_\Omega f(x)v(x)dx\,, x \in \Omega, \qquad (3)$$

де $p_h(x) \in \mathcal{P}_h$ – шукана функція, $\mathcal{P}_h = span\{\phi_0, \dots, \phi_{n_c}\}$.

$\phi_i, i = \overline{1, n_c}$ – різномасштабні базисні функції, які визначаються як розв'язок

$$-div(k(x)\nabla\phi_i) = 0 \text{ в } K_j, \phi_i = \phi_i^0 \text{ на } \partial K_j, i = \overline{1, n_c}, j = \overline{1, m_c}, \qquad (4)$$

де K_j, $j = \overline{1, m_c}$ – прямокутні скінченні елементи, ϕ_i^0 – кусково-лінійні базисні функції методу скінченних елементів [2].

Двовимірні ізопараметричні апроксимації

Для розв'язання (4) використаємо класичний МСЕ з двовимірними ізопараметричними апроксимаціями [3].

Варіаційне формулювання для задачі (4)

$$\int_{K_j} div\big(k(x)\nabla u(x)\big)\cdot v(x)dx = 0, x \in \Omega, \qquad u, v \in V. \qquad (5)$$

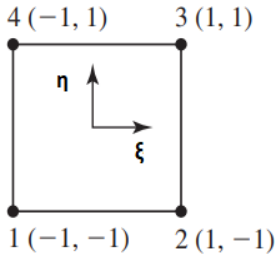


Рисунок 1 Скінченний елемент Ω^*у вигляді квадрата з чотирма вузлами

Побудуємо на вузлових точках квадрата білінійну систему базисних функцій

$$\phi_i^0(\xi,\eta) = \frac{1}{4}(1+\xi_i\xi)(1+\eta_i\eta), \qquad i = \overline{1,4},$$

де ξ_i, η_i – координати i-ї вузлової точки.

Відображення квадрата Ω^* на деякий чотирикутник Ω_e за допомогою перетворень

$$x_1 = \sum_{i=1}^{4} x_1^{(i)}\phi_i^0(\xi,\eta), x_2 = \sum_{i=1}^{4} x_2^{(i)}\phi_i^0(\xi,\eta),$$

де $x_1^{(i)}, x_2^{(i)}$ – координата i-ї точки чотирикутника в площині x_1x_2.

Візьмемо в формулі (5) $u, v = \phi_i^0 \in V, i = \overline{1,n}$ і розіб'ємо інтеграл по області на суму інтегралів по скінченних елементах, отримаємо

$$\sum_{r=1}^{m}\int_{\Omega_r} div\left(k(x)\nabla\phi_i^0(x)\right)\cdot\nabla\phi_j^0(x)dx = 0\,, i,j = \overline{1,n}. \qquad (6)$$

Після переходу до системи координат $\xi\eta$, отримаємо, що

$$\int\limits_{\Omega_r} div\left(k(x)\nabla\phi_i^0(x)\right)\cdot\nabla\phi_j^0(x)dx$$

$$=\int\limits_{-1}^{1}\int\limits_{-1}^{1} div\left(k(\xi,\eta)\nabla\phi_i^0(\xi,\eta)\right)\cdot\nabla\phi_j^0(\xi,\eta)|J|d\xi d\eta,$$

де $|J|-$ Якобіан переходу.

Тоді наближений розв'язок (4), а саме вигляд різномасштабних базисних функцій, будемо шукати як

$$\phi_i(\xi,\eta)=\sum_{j=1}^{n}\widetilde{\phi_{\iota,j}}\,\phi_j^0(\xi,\eta),$$

де $\widetilde{\phi_\iota}$ – розв'язок системи лінійних рівнянь (6).

Взявши в формулі (3) $v=\phi_i^0\in V, i=\overline{1,n_c}$ і розбивши інтеграл по області на суму інтегралів по скінченних елементах, отримаємо

$$\sum_{r=1}^{m_c}\int\limits_{K_r} k(x)\nabla\phi_i(x)\cdot\nabla\phi_j^0(x)dx=\sum_{r=1}^{m_c}\int\limits_{K_r} f(x)\phi_j^0(x)dx\,, i,j=\overline{1,n_c}. \qquad (7)$$

В формулі (7) також треба перейти до системи координат $\xi\eta$ і отримаємо інтеграли в новій системі координат по скінченних елементах для лівої і правої частини (7) відповідно

$$\int\limits_{-1}^{1}\int\limits_{-1}^{1} k(\xi,\eta)\left(\frac{\partial\phi_i(\xi,\eta)}{\partial x_1}\frac{\partial\phi_j^0(\xi,\eta)}{\partial x_1}+\frac{\partial\phi_i(\xi,\eta)}{\partial x_2}\frac{\partial\phi_j^0(\xi,\eta)}{\partial x_2}\right)|J|d\xi d\eta,$$

$$\int\limits_{-1}^{1}\int\limits_{-1}^{1} f(\xi,\eta)\phi_j^0(\xi,\eta)|J|d\xi d\eta.$$

Розв'язок задачі $(1)-(2)$ будемо шукати у вигляді

$$p_h(x_1,x_2)=p_h(\xi,\eta)=\sum_{i=1}^{n_c}p_i\phi_i(\xi,\eta),$$

де $p_i, i=\overline{1,n_c}-$ розв'язок системи лінійних рівнянь (7).

Для чисельного обчислення інтегралів по обох змінних можна використати складену формулу лівих прямокутників.

Для реалізації різносштабного МСЕ використано MATLAB [4].

Використані джерела

[1] Efendiev Y., Hou T. Multiscale finite element methods. Theory and application. — New York: Springer, 2009. — 234 p. — (Surveys and Tutorials in the Applied Mathematical Sciences; vol. 4).

[2] Савула Я. Г. Числовий аналіз задач математичної фізики варіаційними методами. — Львів: видавничий центр ЛНУ імені Івана Франка, 2004. — 221 с

[3] Hutton David V. Fundamentals of Finite Elements Analysis / David V. Hutton. — 1st ed, 2004. — 494 p.

[4] Офіційний сайт документації MATLAB. [Електронний ресурс]. – Режим доступу: https://www.mathworks.com/help/matlab/

IMPROVE THE FREQUENCY RESOLUTION OF A 3D PHYSICALLY BASED SIMULATION OF MAJOR EARTHQUAKE

Iryna Kovalchyk
Ivan Franko National University of Lviv
Faculty of Applied Mathematics and Informatics,
University of L'Aquila
Department of Information Engineering, Computer Science and Mathematics
iryna.kovalchyk@student.univaq.it

The purpose of this study is to analyse the propagation of seismic waves on three-dimensional domains by combining techniques of numerical analysis, high-performance computing and AI. The main idea is to simulate complex seismic phenomena on a digital twin of the real geological domain, containing, for example, accurate reconstructions of alluvial basins, topography, passive fault structures, and any other irregularities of the subsoil that can modify the propagation of seismic waves. These simulations are usually very expensive from a computational point of view, and the cost increases considerably by reducing the mesh size. Therefore, the frequency resolution range is up to 1.5-2.5 Hz, depending on the selected numerical methods, on the mesh size and on the mechanical properties of the subsoil. This limit can be overcome by combining numerical simulations with artificial intelligence techniques.

3D physics-based earthquake simulations

For the last two decades there has been a big step forward in field of development of high-order numerical methods, which helps to simulate seismic propagation on real data of tectonic and geo-morphological conditions. In 2013 Mazzieri et al. in [MSGS13] presents a new tool to simulate seismic events on complex geological domains. In a way, to do that, there were some limitation to overcome. By it we could consider a number of factors such as the propagation path through complex geological structures, coupled effects of the seismic source, localized superficial irregularities (which includes man-made infrastructures and/or alluvial basins). Main purposes of that work is development of flexible, certified, highly accrue numerical software package for elastodynamics type of problems.

Due to improvement of limitation of capabilities there was made some modifications to the SE method. It was done by Discontinuous Galerkin (DG) technique (a non-conforming discretization approach). DGSE methods, in their

general formulation, gives chance to work with locally varying mesh size (h-adaptivity) and non-uniform polynomial degree distribution (N-adaptivity).

SPEED (Spectral Elements in Elastodynamics with Discontinuous Galerkin http://mox.polimi.it/it/progett) represents a very useful and versatile tool to simulate past earthquakes and generate future seismic scenarios, see for example [PEMS16, EDGS+17, IMÖ+20, AMM+20].

The results provided by SPEED as well as by other similar tools (such as SPECFEM3D_Cartesian [KLT+04]) have a frequency resolution limited to a few Hz, but for the assessment and mitigation of seismic damage it is instead important to go to higher frequencies. One possible solution could be to reduce the mesh size, with an exponential increase in the computational cost. To overcome this problem Paolucci et al. [PGI+18] propose an innovative approach based on the combination of physically based 3D numerical simulation and AI techniques, that will be briefly describe in the following section.

An AI tool for broadband earthquake ground motion

The main idea of which use Artificial Neural Network (ANN) to generate broadband ground motions for a seismic scenario and potentially overcome the limitations of Ground-motion prediction equations (GMPEs), often used to obtain ground motion scenarios. GMPEs has many shortcomings such as provide just peak value; constant growth of parameters; empirical coefficients vary when calibration data-sets are updated; neglecting the site-specific features; point-wise prediction does not work correctly (not precise enough). In more details the authors generate Broadband waveforms (BB) ground motions, coupling the results of Physics-Based Simulation (PBS) for a specific earthquake ground-motion scenario with the predictions of an ANN, overcoming some of the main issues of Hybrid Broadband approach (HYB) modeling. This innovative approach has been tested on the 2012 Po Plain earthquake (Mw 6.0) with satisfactory results in [PGI+18] and on other recent seismic events such as L'Aquila 2009 in [PSV21]

What is done and what will be implemented

We have already done reviewing of the literature about this topic, as well as latest research in this field. We are aiming to introduce the 3D-domain simulation of earthquake. As a case study we consider the territory of L'Aquila that its history it has been hit by many strong earthquakes. The last one in 2009 by a strong earthquake of Mw 6.1 which partially destroyed the city and caused more than 300 deaths.

One of the most effective approaches for 3D domain seismic wave propagation is Spectral Element method (SE). It has ability to provide highly

accurate and fast solutions also its native orientation towards high performance parallel computing. The main point of SE method is the finite dimensional space is made high order piece-wise Lagrangian polynomials. Implementation of practical part will be done by using software called Cubit (https://coreform.com/) and programming language Python. We plane to do data analysis on data which we will get from our model to compere results depends on which exact mesh will be stimulated to simulate earthquake.

This kind of research are extremely important for understanding behaving of system in situation of earthquake. Moreover, precise modeling of it could help not just in seismic meaning also it is helpful for civil engineering.

References

[AMM+20] Paola F Antonietti, Ilario Mazzieri, Laura Melas, Roberto Paolucci, Alfio Quarteroni, Chiara Smerzini, Marco Stupazzini, et al. Three-dimensional physics-based earthquake ground motion simulations for seismic risk assessment in densely populated urban areas. 2020.

[EDGS+17] L. Evangelista, S. Del Gaudio, C. Smerzini, A. d'Onofrio, G. Festa, I. Iervolino, L. Landolfi, R. Paolucci, A. Santo, and F. Silvestri. Physics-based seismic input for engineering applications: a case study in the Aterno river valley, Central Italy. Bull. Earth. Eng., 15(7):2645–2671, 2017.

[IMÖ+20] M. Infantino, I. Mazzieri, A. G. Özcebe, R. Paolucci, and M. Stupazzini. 3D physics-based numerical simulations of ground motion in Istanbul from earthquakes along the marmara segment of the North Anatolian Fault. Bull. seism. Soc. Am., 110(5):2559–2576, 2020.

[KLT+04] Dimitri Komatitsch, Qinya Liu, Jeroen Tromp, Peter Suss, Christiane Stidham, and John H Shaw. Simulations of ground motion in the los angeles basin based upon the spectral-element method. Bulletin of the Seismological Society of America, 94(1):187–206, 2004.

[MSGS13] I. Mazzieri, M. Stupazzini, R. Guidotti, and C. Smerzini. SPEED: SPectral Elements in Elastodynamics with Discontinuous Galerkin: A non-conforming approach for 3D multi-scale problems. Int. J.Numer. Method Eng., 95(12):991–1010, 2013.

[PEMS16] R. Paolucci, L. Evangelista, I. Mazzieri, and E. Schiappapietra. The 3D numerical simulation of near-source ground motion during the Marsica earthquake, central italy, 100 years later. Soil Dyn. Earthq. Eng., 91:39 – 52, 2016.

МЕТОДИ ТИПУ НЬЮТОНА ДЛЯ РОЗВ'ЯЗУВАННЯ ОБЕРНЕНОЇ ЗАДАЧІ НА ВЛАСНІ ЗНАЧЕННЯ

Юлія Жорняк

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

Yuliia.Zhorniak@lnu.edu.ua

Обернена матрична задача на власні значення – це задача знаходження невідомих параметрів деякої матриці за заданим скінченним набором спектральних даних. Ці дані можуть містити лише часткову або повну інформацію про власні значення або власні вектори. Задача полягає в тому, що потрібно побудувати матрицю, яка зберігає задані спектральні властивості.

Розглянемо загальну обернену задачу на власні значення.

Нехай $A_i = \{a_{jk}^i\}$ – задані комплексні матриці, $i = 1, \dots, n$ і $\lambda_1, \dots \lambda_n \in C$ – задані числа.

Знайти такі параметри $p_1, p_2 \dots p_n \in C$, що матриця

$$A(p) = A_0 + \sum_{i=1}^{n} p_i A_i, \qquad (1)$$

має власні значення $\lambda_1, \dots, \lambda_n$.

У [1] запропоновано метод Ньютона для розв'язування задачі (1), який обчислює нулі функції

$$F(p) = \begin{bmatrix} \det(A(p) - \lambda_1 I) \\ . \\ . \\ . \\ \det(A(p) - \lambda_n I) \end{bmatrix}, \qquad (2)$$

і вимагає на кожному кроці n обертань матриць $(A(p) - \lambda_1 I)$, i=1, 2, . . n, I – одинична матриця.

На кожному кроці методу Ньютона потрібно обчислювати n визначників матриць $(A(p) - \lambda_1 I)$, i=1, 2, . . n, а також матрицю часткових похідних від цих визначників.

У праці [4] запропоновано ефективний чисельний алгоритм обчислення точних похідних від детермінанта матриці, який ґрунтується на LU-розкладі матриці. Цей підхід дає змогу обчислити не тільки першу похідну детермінанта матриці, а й другу, що дає змогу застосовувати обчислення нулів детермінанта.

Нехай потрібно обчислити $f(\alpha) = \det\mathbf{D}(\boldsymbol{\alpha})$ і $f'(\alpha) = (\det\mathbf{D}(\boldsymbol{\alpha}))'$. Тоді для матриці $\mathbf{D}(\boldsymbol{\alpha})$, для якої за фіксованого значення α головні мінори всіх порядків відмінні від нуля, справедливе подання

$$\boldsymbol{D}(\boldsymbol{\alpha}) = \boldsymbol{L}(\boldsymbol{\alpha})\boldsymbol{U}(\boldsymbol{\alpha}), \tag{3}$$

а для матриці $\boldsymbol{B}(\boldsymbol{\alpha}) = (\boldsymbol{D}(\boldsymbol{\alpha}))'$

$$\mathbf{B}(\boldsymbol{\alpha}) = \mathbf{M}(\boldsymbol{\alpha})\mathbf{U}(\boldsymbol{\alpha}) + \mathbf{L}(\boldsymbol{\alpha})\mathbf{V}(\boldsymbol{\alpha}). \tag{4}$$

Тут $\mathbf{L}(\boldsymbol{\alpha})$ – нижня трикутна матриця з одиничними діагональними елементами, а $\mathbf{U}(\boldsymbol{\alpha})$ – верхня трикутна матриця, $\mathbf{M}(\boldsymbol{\alpha}) = \mathbf{L}(\boldsymbol{\alpha})'$, $\mathbf{V}(\boldsymbol{\alpha}) = \mathbf{U}(\boldsymbol{\alpha})'$. Тоді

$$\boldsymbol{f(a)}=\prod_{i=1}^{n} \boldsymbol{u_{ii}}$$

$$\boldsymbol{f(a)'}=\sum_{k=1}^{n} \boldsymbol{v_{kk}} \prod_{i=1, i\neq k}^{n} \boldsymbol{u_{ii}} \qquad \textbf{(5)}$$

Запишемо алгоритм ітераційного методу Ньютона для розв'язування оберненої задачі на власні значення.

1. Задаємо початкове наближення $p^{(0)}$.
2. Для $k = 0,1,\ldots$ до досягнення точності виконати
3. Обчислюємо $(A(p^{(k)}) - \alpha_i I), i = 1, \ldots, n$.
4. Обчислюємо LU-розклади (3), (4) та $F_i(p^{(k)})$ і $\frac{\partial F_i(p^{(k)})}{\partial p_j}$ за (5).
5. Формуємо матрицю $J(p^{(k)})$ і вектор $F(P^{(k)})$.
6. Обчислюємо $p^{(k+1)}$, розв'язуючи систему
   $J(p^{(k)})(p^{(k+1)} - p^{(k)}) = -F(p^{(k)})$.

В методі Ньютона на кожній ітерації потрібно обчислювати матрицю часткових похідних. Для зменшення обчислень можна F' перераховувати один раз на m кроків. Поєдання цього підходу з МПІ призводить до ітераційного методу з порядком збіжності $m + 1$ [3]

$$\begin{aligned} &y^{(l+1)} = y^{(l)} - F'(x^{(k)})^{-1}F(y^{(l)}), l = 1, \ldots, m-1, y^{(1)} = x^{(k)}, \\ &x^{(k+1)} = T(x^{(k)}) = x^{(k)} - F'(x^{(k)})^{-1}\sum_{l=1}^{m} F\,(y^{(l)}). \end{aligned} \tag{6}$$

Кордеро та ін. [2] представили метод четвертого порядку, який вимагає обчислення двох вектор-фунцій F, двох матриць F' і однієї інверсії матриці. Він задається формулою

$$y^{(k)} = \phi_1^{(2)}(x^{(k)}) = x^{(k)} - [F'(x^{(k)})]^{-1}F(x^{(k)}),$$

$$x^{(k+1)} = \phi_1^{(4)}(x^{(k)}, y^{(k)}) = y^{(k)} - (2[F'x^{(k)}]^{-1} - [F'x^{(k)}]^{-1}F'(y^{(k)})[F'(x)^{(k)}]^{-1})F(y^{(k)}),$$

(7)

Наведено результати застосування описаних вище трьох методів для розв'язування оберених задач на власні значення. Нехай маємо матриці

$$A_0 = \begin{pmatrix} 2 & 1 \\ 3 & 4 \end{pmatrix}, \qquad A_1 = \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix}, \qquad A_2 = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}.$$

Потрібно знайти p_1 та p_2 такі, щоб матриця

$$A(p) = A_0 + p_1 A_1 + p_2 A_2$$

мала власні значення $\lambda_1 = 1$ та $\lambda_2 = -1$.

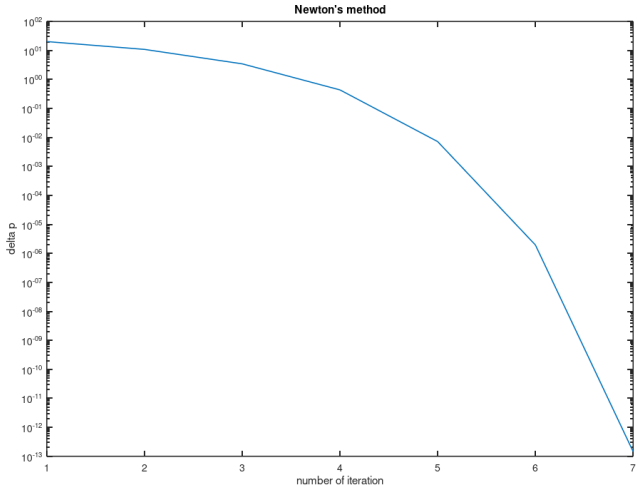


Рис 1. Метод Ньютона: зміна $\delta p^{(k)}$ на кожній ітерації

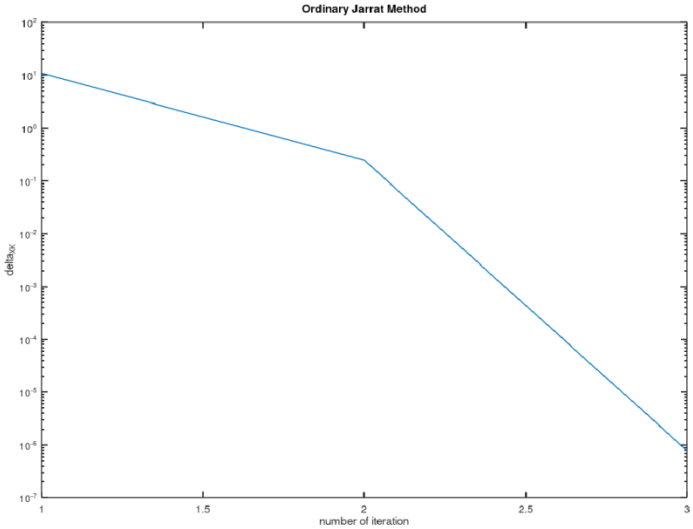


Рис 2. Модифікований метод Ньютона: зміна $\delta p^{(k)}$ на кожній ітерації

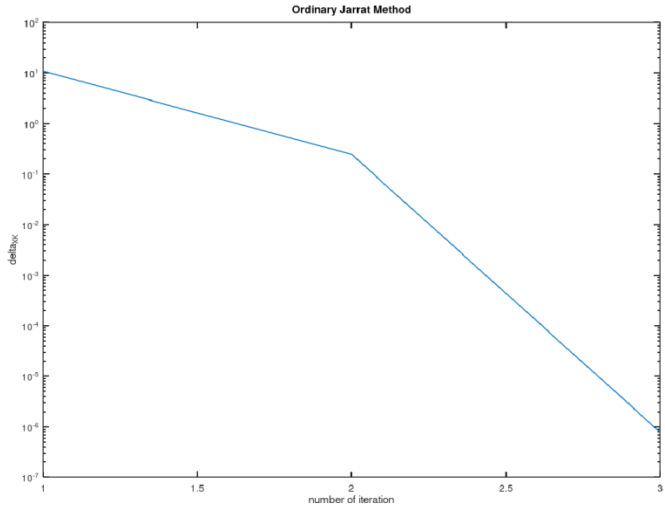


Рис 3. Метод Джаррата: зміна $\delta p^{(k)}$ на кожній ітерації

Табл 1. Результати отримані для початкового наближення $p^{(0)} = (0; 0)$

	k	$\| \delta p^{(k)} \|$	$\| F(p^{(k)}) \|$	$p^{(k)}$
метод Ньютона	9	8.3925e-17	4.4409e-16	-1.062746066806228 -1.291502622129181
мод. метод Ньютона	3	5. 6382e-17	4. 4409e-16	-1.062746066806228 -1.291502622129181
метод Джаррата	3	3. 7046e-08	7. 8413e-07	-1.062746066806213 -1.291502622129192

Список літератури

[1] Biegler-Konig F. W. A Newton iteration process for inverse eigenvalue problems // Numer. Math. – 1981. – 37, No. 3. – P. 349–354.

[2].Cordero, A. , Martínez, E. , Torregrosa, J. R. Iterative methods of order four and five for systems ofnonlinear equations. J. Comput. Appl. Math. 231, 541–551 (2009)

[3] Annals of West University of Timisoara – Mathematics and Computer Science **57**(1), 13–22 (2019).

[4] Подлевський Б. Обчислення точних похідних детермінанта матриці / Б. Подлевський // Вісн. Львів. ун-ту. Сер. прикл. матем. та інф. – 2013. – Вип. 20. – С. 42-48.

ПРОТОТИП САЙТУ НА ОСНОВІ СТАТИЧНИХ ДОСЛІДЖЕНЬ ТА ВИБОРУ ОПТИМАЛЬНОЇ СТРАТЕГІЇ

Любов-Василина Вовк

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

Vasylyna.vovk2002@gmail.com

Світ давно став цифровим і все більше бізнесів створюють сайти для представлення в інтернеті, але часто ці сайти не приносять ніякого результату. Структури цих застосунків давно застаріли, а блоки не несуть ніякого цільового завдання.

Метою цього проекту є показати, що правильно складений сайт є відображенням бізнесу в інтернеті, його розширенням, створенням реклами, збільшенням продажу наявних товарів і послуг та який приводить клієнтів через продумані стратегічні рішення, додає впізнаваність бренду і відображає преміальність даного салону, для якого ми розробляємо сайт.

І далеко не просто застосування всім відомих HTML, CSS, JS для розробки сайту.

Я вирішила розробити сайт, провести аналітичне дослідження та створити стратегічний прототип, а також упакувати це у сучасний стильний унікальний дизайн, який не залишить увагу користувача, та зверстати його на конструкторі сайтів Readymag, вставляючи частки власного коду для більшого функціоналу.

Перш за все, був проведений аналіз цільової аудиторії, адже від цього напряму впливає вигляд сайту.

В ході проведення дослідження це – дівчата та жінки віком від 25-50 років, для них важлива статусність салону, рівень сервісу та професіоналізм майстрів, вони готові платити дорого за послуги.

Далі ми проаналізували конкурентне середовище, виявили їх недоліки та переваги, за рахунок цього аналізу наш салон буде виділятися на фоні конкурентів.

Після вивчення конкурентів я розробила карту сайту, щоб вибудувати ефективну стратегію впливу на відвідувача та показала шлях клієнта.

За допомогою двох формул (ACCA і AIDA) розробила шлях клієнта і карту сайту.

AIDA

Саме ця формула, яка залишається актуальною вже більше століття, стала основою для сучасного маркетингу і все ще залишається інструментом, що активно використовується в середовищі фахівців з реклами.

За цією класичною моделлю вже більше ста років вибудовуються рекламні звернення, що проводять ЦА до необхідних дій через 4 основні кроки:

Уваги (Attention) → Цікавості (Interest) → Бажання (Desire) → Дії (Action)

У спрощеному вигляді послідовність дій за формулою AIDA можна побачити на нашому графіку:

ACCA

Як і перша схема, ACCA вказує на необхідність проведення потенційного покупця за 4 станами для отримання необхідного ефекту як цільової дії:

Уваги (Attention) → Розуміння (Comprehension) → Переконання (Conviction) → Дії (Action)

Не важко помітити, що перший та останній кроки в цих схемах аналогічні, але все, що між ними, має серйозні відмінності.

За допомогою графіки розглянемо, як саме модель ACCA діє на ЦА на нашому сайті:

ACCA зараз є набагато більш ефективною формулою, оскільки в її рамках наводяться конкретні аргументи без зайвих маніпуляцій. Це дозволяє

вагомо переконувати сучасного прискіпливого клієнта у правильності прийнятих рішень. Звичайно, все це стосується тільки тих пропозицій, для яких є достатня кількість реальних аргументів.

Але ми вирішили застосувати 2 стратегії, аби максимально вплинути на потенційних відвідувачів.

Далі був розроблений прототип та дизайн у графічному редакторі, враховуючи всі наші аналітичні дослідження.

Для реалізації всього вищезазначеного я обрала платформу Readymag, тому що вона має ряд переваг:

по-перше, ми можемо створювати унікальний дизайн, а не використовувати стандартні блоки;

по-друге, є можливість додати власний код для збільшення функціоналу;

по-третє, платформа інтегрована з різними сервісами, що полегшує збір та обробку інформації, такі як підключення тегів для пошукових систем і соціальних мереж та підключення до Google Аналітики.

У результаті ми отримуємо грамотно пропрацьований сайт, який відповідає найбільш працюючим маркетинговим стратегіям, орієнтований на цільову аудиторію із сучасним та актуальним дизайном та потрібним функціоналом, що допоможе бізнесу привести клієнтів та збільшити впізнаваність.

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ НАДІЙНОГО ТА ЕФЕКТИВНОГО ШИФРУВАННЯ ДАНИХ

Данило Брозь
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
daniel.bros1169@gmail.com

Вступ

Шифрування – це по суті код, який використовується для приховування вмісту повідомлення або даних, які ми надсилаємо,отримуємо чи зберігаємо.

Я хочу дослідити якомога більше шифрів,зрозуміти принцип їх роботи, спробувати знайти метод їх дешифрування чи взлому, а також визначитись з найбільш надійним шифром.

Сучасне шифрування

1) Симетричне шифрування використовує один криптографічний ключ для шифрування та дешифрування даних і це робить його простим в процесі.Найстарішим симетричним методом шифрування є DES. Він перетворює 64-бітні блоки даних відкритого тексту у зашифрований. Це відбувається завдяки поділу на два різних 32-бітні блоки та застовується окремо до кожного.

Згодом з`явився алгоритм AES, який вважається найкращим та найпопулярнішим на даний момент. Він працює методом підстановки та перестановки.Спершу, незашифровані дані перетворюються в блоки, а потім шифрування виконується з використанням ключа.

2) Асиметричне шифрування включає декілька ключів для шифрування та дешифрування даних. Цей тип забезпечує більшу безпеку,наявність обох ключів, один з яких загальнодоступний, а інший – закритий, гарантує, що дані будуть захищеними від кібератак та аутентифікацію.

Алгоритм RSA є дуже ефективним оскільки йому притаманний метод «первинної факторизації», масштабованість, ключі можуть бути різної довжини та адаптивність.

Застосування алгоритму RSA

1-й етап: Генерація ключів

- Обираємо два простих різних числа: x та y
- Обчислюємо добуток: n=x*y
- Обчислюємо функцію Ейлера: $\varphi(n)=(x-1)*(y-1)$
- Обираємо відкриту експоненту: e=3

- Обчислюємо секретну експоненту: d= e^{-1} (mod φ(n))
- Публікуємо відкритий ключ: {e,n}
- Зберігаємо секретний ключ: {d,n}

2-й етап: Шифрування

- Обираємо текст для
- шифрування: m
- Обчислюємо шифротекст: c= E(m) = m^e mod n

3-й етап: Розшифрування

- Обчислюємо вихідне повідомлення: m= D(c)= c^d mod n

Алгоритм AES-256.

Цей алгоритм проходить через декілька раундів шифрування і кожен раунд включає в себе наступні кроки:

1. Розділюємо дані на блоки
2. Ключове розширення
3. Добавляємо круглий ключ
4. Замінюємо байти
5. Зсуваємо ряди
6. Перемішуємо стовпці
7. Знову додаємо круглий ключ
8. Робимо це знову
9. Після останнього раунду алгоритм пройде ще один раунд, але в цьому наборі алгоритмів виконає усі кроки окрім 6-го, оскільки цей крок є беззмістовним.

Висновок

Отже, зараз найчастіше використовується симетричне та асиметричне шифрування. Проаналізувавши їх, я зрозумів що симетричне шифрування набагато швидше, в той час, асиметричне - надійніше. Тому, найкращим варіантом для захисту даних є змішаний тип шифрування: шифруємо початкові дані симетричним методом, а приватний та публічний ключ асиметричним.

Літературні першоджерела

1. Навчально-методичний посібник Л.Я. Глинчук «Криптологія»
2. Навчальний посібник Т.В. Бабенко, Г.М. Гулак «Криптологія у прикладах, тестах і задачах»
3. https://www.websiterating.com/ru/cloud-storage/what-is-aes-256-encryption/
4. https://proverkassl.com/book_ssl_history.html
5. https://hostpro.ua/wiki/ua/security/encryption-types-algorithms
6. https://experience.dropbox.com/uk-ua/resources/what-is-encryption

MICROSCOPIC SIMULATION OF ROAD TRAFFIC

Abayomi Dennis Epebinu
Ivan Franko National University of Lviv,
University of L'Aquila

Abstract

Microscopic simulation models have been widely used as tools to investigate the operation of road traffic and different intelligent transportation systems applications. The conformity of microscopic simulation tools depends on the driving behaviour models that they implement. We desire to model and simulate road traffic. Naturally, we had as one goal, the better understanding of road traffic and we modelled it using cellular automata (CA). Our model explains traffic occurrences as well as possible and simulate real traffic.

Keywords: Cellula Automata, Traffic, Microscopic, Simulation.

Introduction

Traffic modelling is aimed to accurately recreate traffic as observed and measured on roads. Traffic modelling assumed the appearance of a traffic system without replicating. we are no longer interested completely in the standard value of significant traffic parameters for a roadway such as the velocity v in km/h, the flow f in veh/h or the density ρ in veh/km. To control the flow, we amount the number of vehicles transitory by a check point per unit time. For the density, we consider the number of vehicles per leg on a domination section. At the microscopic simulation, we want to determine the traffic down to the individual traffic partaker to “microscopic” precision in mandate to consider the individual behaviour. From the point of assessment of the individual driver, this is significant in the case for route planning. This should be as lively as possible and be reliant on the current traffic situation. It must also account for individual possessions such as the maximum speed of each driver’s automobile.

Objective of the research

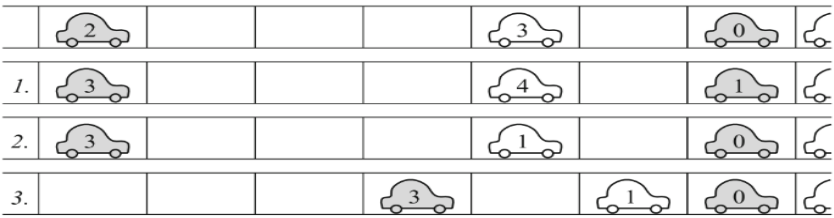
In this research we want to introduce a model that is based on stochastic cellular automata. In its simple form, this model is very modest. However, it models and explains different traffic phenomena and it is very useful in applications. We transfer the concept of cellular automata to the traffic simulation.

Results

There are two central model assumptions that our model shall satisfy, they are:

1. Absence of collision
2. Conservation of vehicle.

Absence of collisions means that two vehicles may never drive within the identical cell within a time step or between two-time steps. This applies, specifically, that a vehicle must be able to decelerate without any time lag. If a car travels at $v = v_{max} = 5$ and, at time step t, arrives in the cell directly behind a non-moving car, then it must be assured that in the next time step it reduces its velocity to $v = 0$ and also stops moving. Hence it holds for the neighborhood connection. For this modest model it is necessary for a vehicle to look ahead the highest step width in the driving direction, i.e., five cells. Let the vehicles be numbered with regard to the driving route, i.e., vehicle I drives with velocity v_i behind vehicle $i + 1$, with velocity $v_{(i+1)}$, let $d_{(i,j)}$ be the distance of vehicle i to vehicle j in driving direction, i.e., the number of cells between i and j. Two cars that stand sprightly one after the other have distance zero. The rules of the change function hold in parallel for all vehicles.



Fig(1). car moving with different velocity.

This movement is with the following CA rule or algorithm.

Update for vehicle i with $j = i + 1$:

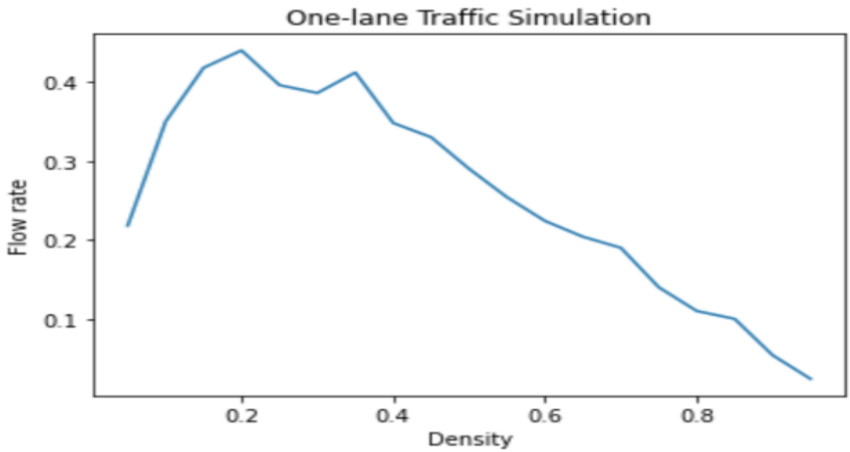
1. Accelerate: $v_i := \min(v_{i+1}, v_{max})$
2. Decelerate: $v_i := d(i, i + 1)$; if $v_i < d(i; i + 1)$
3. Randomize: $v_i := \max(v_{i-1}, 0)$ with probability $p < 1$
4. Move: vehicle i moves v_i cells forward.

Road traffic simulation using cellula automata, what is of interest is that this simple model can actually be simulated. We simulate a one lane unidirectional highway section.

```
.1...3.0.....50..2...0.1...3.1...3...1......4...1.....4...20.0.........5.00......5..10.000.....5...3
1.1..0..1....0.1....30...2..1.1.....3..2.......3..2.....2.0.1.1........0.0.1.......20.100.1......2..
.1..2.1..1....1..2..0.1....2.1..2....1...2.......2..2....1.1.1..2......0..1..2.....0.100.1..2.....1.
...2.1..2.1.....2..20...2...1..2...3...2....3......2...3..1.1..2...3...0....2...3..0.00.1.1..2....1
.2..1..2.1..2.....20.1.....3..2...3..2....3.....4.....3.1.0...2...3...3.1.....2...2.10.10...2....3..
3..2..20...2...3..00..1.....2..2...2...3....3......4..1.10......3...3.1..2......3.10.100.....2.....
.1...200.....2...20.1...2......2...3...3...3.....4...1..10.1.......2..1..2..2.....100.00.1.......3..
3..2.000........300..1....2.......3...3...3....4....3..20.1.1.......1..1...2...3..00.10.1..2........
.1.0.000........00.1..1.....3.......3...3....4...3..1.0.1.1..2......1..1.....3.1.0.100...2..2......
.0.0.00.1.......0.1..2..2.......3.......3....4...3..2.1.1.1..2...3.....2.1.....1.1.1000....1....3...
40..100...2......1..2..2..2........4.......4..2...2.10.0...2...3....4..1..2....1.10000.....1.......
00..000......3.....2..2..2...3..........5...1....3.10.1.1.....3....4.1...2...3..10000.1......2.....
0.1.00.1........3....2.1....3....4........2..1...0.0.1.1..2......4.1..2....3..20000.1..2......2...
0..100...2........3..1..2......4.....5.....1..1...1.1.1.1...2.....0...2...3...200000..1...2.......3
.1.00.1.....3.......2.1....3.......5.....5..2..2.0..1.1.1...2....1.....3...300000.1...2...2.....0
1.10.1..2.......4....0...2......4........5..2..2.10...1.1.1......3...2......300000.1.1.....3..2....
.10.1..2...3.......40......3.......5.....1...2.10.1...10..1........3...3...000000.0..1......2...3.
20.1..2...3...3.....00..........4........5..2.0.0.1..2.0.1...2........2...2.00000.10....2.......3..
0.1..2...3...3...3..0.1.............5......20.0..1..2.1.1..2....3........3.10000.10.1......3.....1.
.1.1....3...3...3..2.1..2................5.0.1.1..1.0..1..2..2......4.....10000.100..1........3...1
1.1..2....2....3..20...2...3.............0..1.1..2.1.1...2..2...3........50000.100.1...2........2..
.1.1....3....3...20.1.....3...3............1..1..20..1..2...2...3....4....0000.100.1..2....3......1.
20...2.....3....30.1..2......2....3...........2.1.00....2...3...3....4....4000.100.1..2...3....3.....
00......3......400...2...3....2......4.......0..100.......3...3....4....40000.00.1..2...3...3.....4.
00..........4..000......3...3....3.......4....1.00.1........3...3.....40000.10.1..2...3...3....4..1
```

Fig(2). Single lane simulations with 25 timesteps.

So, we plot the graph of flow rate to the density for one lane traffic with 25 time steps.



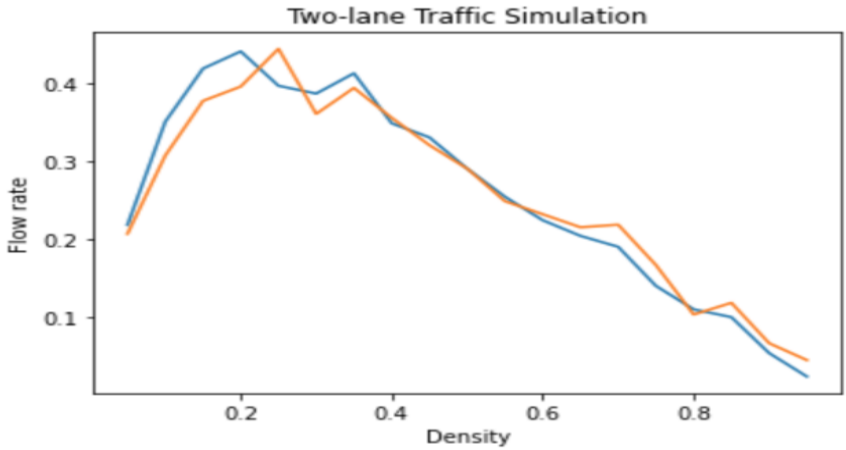
Fig(3). Average flow rate for 25 timestep.

For two lane simulation of the road traffic. For two lanes, the randomization step is very useful and sufficient for the model to demonstrate realistic traffic behaviour.

```
....4.0....4.......3...........2....40...2.......................2.10................4...........2.10
4.....4...1...3...2........1...1.........4.....3...........3.1......1.............5.0.....4..10....
1...0..1.......4......4.........3.00.....2.....................10.1..................4........10.
.....5...3..2....3...3......1...1..............5...3..........1.1......1.............1.1......20.1...
..2.0...10........4........5......100........3..................0.1.1.......................5...00.
...............3....3....4...1....2..............2....3......0...2......2............1..2....0.1..2.
...10...00.............4........5.00.1...........4..............1.1..2.......................2.00.
.3................3.....4..2...2.....3.............3.....4..0.....2.......3...........2...3.0..1...
.....1..00...................5....0.00..1..............5...........1..2...3..................0.0.1
..10..................4...2...3...3......4..............4...2.1......2.........4..........3.10...1..
.20...1.0.1....................3..10.1...2...................5.......1....3....4.................0..1.
...0.....................2....3...3....4.......5............3.1..2......2...........4.......100....1.
.0.1...1.1..2.....................20.1.1.....3.....................5....2......4.....5............1..1
....1.......................3....3....4.....5.......5........1.1....3.....2..............5..00.1....1
10..1...1..2...3................0.1.1..2.......4.....................4...3........5....4.........1..
1....1.........................4...3......5.....5.......5....1.1......3.....3.............20.1.1....
00....2...2...3...3.............0.0...2...3.........5..................3....4.......3......5......2
..2....2.........................3...3........5.....5.....3.0...2........4.....4.........0.1.1.1...
0.1......3...3...3...3...........1.1.....3....4.........5..................4....4.......4.......5.0
....2.....3.........................3....4.........5....4..10......3..........5.....5.....1.1.1.1..
.1..2......2....3...3....4.........1.1.......3......5.........5..............3.....4.......4....10
.......3......4.......................4.....5.......3..1.0.1.........4...........5....4..1.1.1..2
1.1...2.......3...2.....4.....5.....1..2........4.......5..........5.............3.......5......40.
..3.......3.......4.........................5.....5....2.1.1..2............5..........4..2.1.1.1..
0...2....3......3...3......5.....5..2...3...........5......4.........4.............4........5.00.
.....4.......4........5.........................4....3.0..1..2...3............4........2.10..1..2
0......3....4......3....4........5..2...3....4............5......5..........5.............5...0.0.1
..3........5......4........5.......................4.1.1...2..2.....4..............5....100....2.
.1........4......5.....4.....5.....2...3...3.....4.............5.....4...........5...........3.10.0
.3...3..........5.....4.........4.....................1.1..2...2...3.......5.............400.1.....
```

Fig(4): Double lane simulation with 15 timesteps.

We can realise that the rate at which we have traffic jams has reduced due to ability to change lane. So, we plot the graph of flow rate to the density for two lane traffic with 15 timesteps.



Fig(5): Average flow rate for 15 time steps for two lanes.

Conclusion

We have been able to simulate road traffic microscopically with the help ofcellula automata for a single lane and two lanes, we discover that for single,as cars are moving randomly into the lane from left to right with their respective velocities, there experiences a traffic jam and this is due to different behaviour of the drivers.

But in the case of two lanes traffic, the issue of traffic jam is reduced, this is as a result of the fact that drivers can change lane once there is free space in the second lane, the cellula automata helps us to assume that all cars are same with the assumption that the maximum velocity the car can attain is 5m/s.

References

1. K. Nagel and M. Schreckenberg. A cellular automaton model for freeway traffific. Journal de Physique I, 2:2221–2229, December 1992.
2. Rolf H. Möhring. Scheduling under uncertainty: Bounding the makespan distribution. In Helmut Alt, editor, Computational Discrete Mathematics: Advanced Lectures, volume 2122 of Lecture Notes in Computer Science, pages 79–97. Springer, 2001
3. Edward Ott. Chaos in Dynamical Systems. Cambridge University Press, 1993.
4. Alan Taylor and Allison M. Pacelli. Mathematics and Politics: Strategy, Voting, Power, and Proof. Springer, 2nd edition, 2008.
5. Henk C. Tijms. A First Course in Stochastic Models. Wiley, 2 edition, 2003.
6. M. Treiber and A. Kesting. Traffific Flow Dynamics. Springer Berlin Heidelberg, 2013.

DETECTING PARKING OCCUPANCY USING MACHINE LEARNING

Joshua Junior John

Ivan Franko National University of Lviv,

University of L'Aquila

joshuajunior.john1@gmail.com

Abstract

Detecting parking occupancy have a potential to reduce the congestion in crowded areas by providing real-time indications of availability of parking spaces. With the increasing demand for parking space in outdoor environments, low-cost image-based detection methods have become a focus of research and development recently. This study presents a robust parking occupancy detection framework by using a modified Mask RCNN and a custom CNN Model based on transfer learning.

Keywords : Convolutional Neural Networks(CNN), Intersection over Union (IoU), Mask Region Based Convolutional Neural Networks (Mask RCNN)

Introduction

The number of motor vehicles in the world has increased rapidly due to the fast economic and social development. In contrast, the construction of parking lots has been relatively slow, and the problem of parking difficulties has become increasingly prominent. This increasing growth has also brought about stationary traffic and other problems. Stationary traffic refers to the parking of vehicles, including short-term parking due to passengers getting on and off or loading and unloading of goods, and long-term parking in parking lots. The problems of stationary traffic are mainly manifested in the following aspects: insufficient control of land use indicators, planning and layout, fewer parking spaces in public buildings, a severe road occupation occurrence, and fewer parking spaces in parking lots. The above situation demonstrates that the parking problem is very acute. As a result, detecting parking occupancy has risen to the top of the priority list for resolving parking issues.

Objective of Research

This research paper focuses on mitigating the problems associated with finding an available parking space by applying machine learning technique.

Methodology

For this research, we propose two approach. First is the use of MaskRCNN to automatically detect the parking slots and also available parking slots. While the

next is the use of a custom CNN model which is trained with dataset containing empty and occupied slots.

Results

MaskRCNN: This approach autodetect the parking lots region of interest and uses this information for further detect available parking slots in the future as shown in the Figure 1.

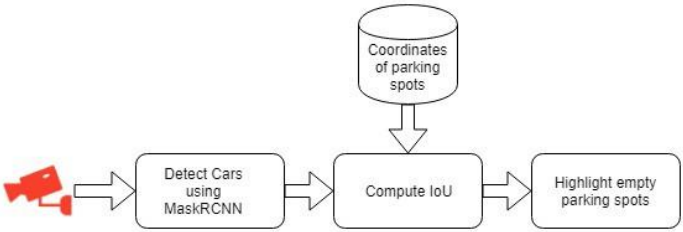


Figure 1 Flow process using M-RCNN approach

On each frame of the input video, we will first use the Mask-RCNN object detection model to detect the cars and their bounding boxes as seen in **Помилка! Д жерело посилання не знайдено.**.

Figure 2: Extracting Bounding boxes using Mask-RCNN

IoU is simply an evaluation metric. As its name suggests it the ratio of the area of overlap and area of intersection. Computing Intersection over Union can, therefore, be determined as shown below.

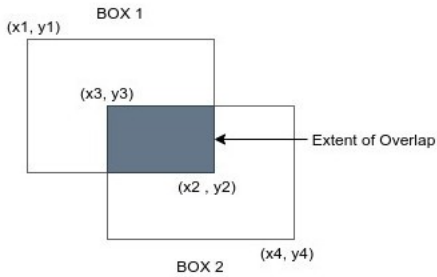


Figure 3: Intersection of two boxes

$$IoU = \frac{Area\ of\ Intersection\ of\ two\ boxes}{Area\ of\ Union\ of\ two\ boxes}$$

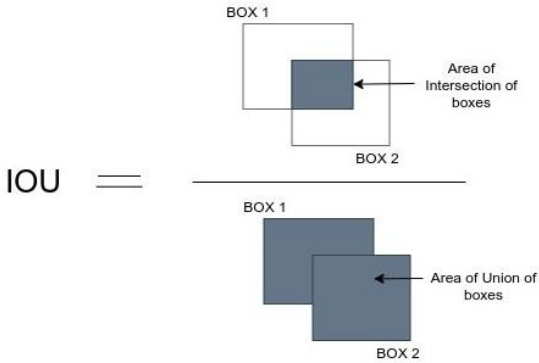


Figure 4 Diagrammatic Representation of the formula to calculate IOU

We then compute the IoU on each pair of the bounding boxes and parking spot coordinates. If the IoU value for any parking spot is greater than a certain threshold, we will consider that parking spot as occupied.

Our model was tested with different IoU thresholds against accuracy and seen below.

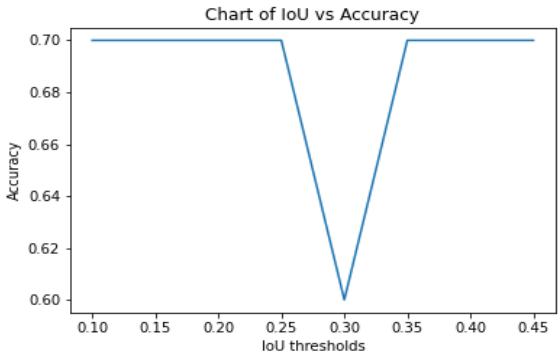


Figure 5 IoU vs Accuracy

Figure 6 Result Using Mask RCNN

Figure 6 shows how well using the Mask RCNN approach performs for a mini parking lot.

Custom CNN Model: For this approach we created a CNN model leveraging on the VGG16 Net to perform binary classification as seen in Figure 7. In order to use the VGG16 Net base mode, we unfreeze the top layer and fused our layer having

our desire output. We added two dense layer and finally the output layer with a sigmoid activation function.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 vgg16 (Functional)          (None, 1, 1, 512)         14714688

 flatten (Flatten)           (None, 512)               0

 dense (Dense)               (None, 50)                25650

 dense_1 (Dense)             (None, 20)                1020

 dense_2 (Dense)             (None, 1)                 21

=================================================================
Total params: 14,741,379
Trainable params: 26,691
Non-trainable params: 14,714,688
```

Figure 7 CNN Architecture using VGG16Net (Transfer Leraning)

$$s(x) = \frac{1}{1 + e^{x}}$$

Signmoid activation function

Our model predicts if a marked slot is empty or occupied. Marking of parking spots was done using python script as shown in Figure 8

Figure 8 Marking of Parking Slots

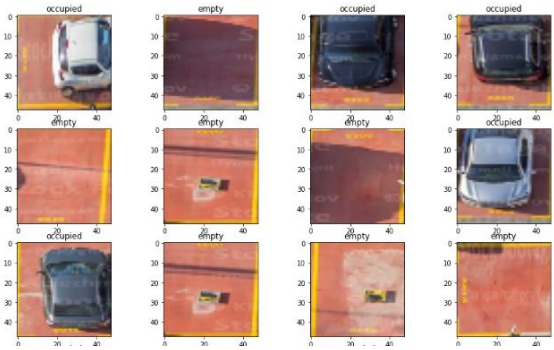


Figure 9 Extracted Slots

The extracted slots are then passed through the model for training and validation. Figure 10 and Figure 11 plots model training and validation accuracy, loss and confusion matrix respectively.

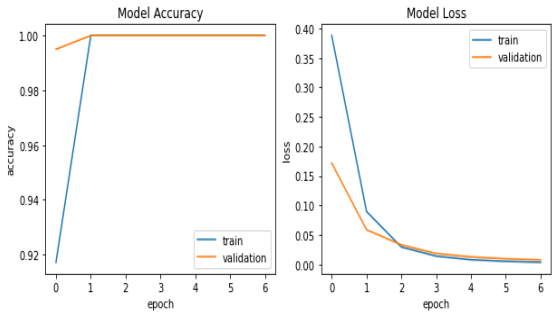


Figure 10 Training and Validation Accuracy and Loss

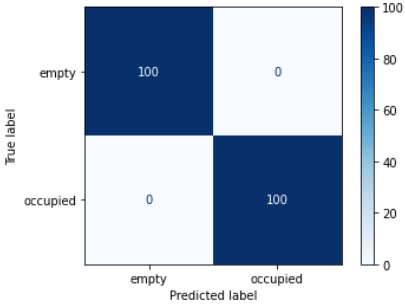


Figure 11 Confusion Matrix

We can see that our model performed well. We then perform test on a parking lot to see how well it performs

Figure 12: Parking Occupancy Detection

Conclusion

This research developed a system for detecting parking space occupancy using machine learning through video data analysis from the outdoor parking area. We have been able to demonstrate our approaches in detecting parking occupancy. The first was using Mask RCNN, this method is suitable for camera positioned at an angle. This involved two main stages; the first stage is identifying the region of Interest by marking the parking position on the image of a full parking lot. The second stage is feature extraction based on the region of interest (ROI) of each marking on the parking area. Experiments for parking space detection are evaluated using the accuracy of IoU. The second proposed idea was using a custom trained CNN which predicts the status of a parking slot. This method works best for cameras positioned directly above the parking lot. Both methods proposed in this research works well during all weather conditions and their implementation provides an accurate automatic parking space occupancy detection and can reduce unnecessary maintenance installation costs.

References

1. T. Thomas and T. Bhatt, “Smart Car Parking System Using Convolutional Neural Network”, In: Proc. of Int. Conf. Inven. Res. Comput. Appl. ICIRCA 2018, No. Icirca, pp. 172–174, 2018.
2. J. Ahmad, Z. Lewis, P. Duraisamy, and T. McDonald, “Parking Lot Monitoring using MRCNN”, In: Proc. of 2019 10th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2019, pp. 10–13, 2019

МОДЕЛЬ ДИФУЗІЙНОГО ПРОЦЕСУ З НАПІВМАРКОВСЬКИМИ ПЕРЕКЛЮЧЕННЯМИ

Мар'яна Шмуль

Науковий керівник: проф. Чабанюк Ярослав Михайлович

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

mariana.shmul@lnu.edu.ua

Вступ

Дифузія, як фізичне явище - це процес взаємного проникнення молекул або атомів однієї речовини поміж молекули або атоми іншої, що зазвичай призводить до вирівнювання їх концентрацій у всьому займаному об'ємі.

У теорії ймовірностей і математичній статистиці процес дифузії є розв'язком стохастичного диференціального рівняння та розглядається як неперервний марковський процес з неперервним часом і майже напевно неперервними траєкторіями вибірки.

Перевагою випадкових величин і стохастичних процесів є те, що їх можна використовувати для опису того, що відбувається навколо нас у світі.

Ми пропонуємо розглянути розв'язок стохастичного диференціального рівняння для дифузійного процесу з напівмарковськими переключеннями, тобто з додатковими випадковими впливами на систему на певних випадкових інтервалах. Очевидно, що ці розв'язки будуть приймати випадкові значення.

Постановка задачі

Дифузійний процес – це розв'язок такого стохастичного диференціального рівняння[2]:

$$dX(t) = a(X(t), t)dt + b(X(t), t)dW(t), \quad t \geq 0, \tag{1}$$

де $a(x, t)$ – коефіцієнти переносу, $b(x, t)$ - коефіцієнт дифузії, $W(t)$ - Вінерівський процес.

Ми розглядаємо один із прикладів дифузійного процесу, а саме рівняння виду:

$$dX(t) = \mu\, X(t)dt + \sigma\, X(t)dW(t), \quad t \geq 0, \qquad \mu,\ \sigma - const. \tag{2}$$

Для дифузійного процесу (2) введемо модифікацію, що моделює ціну акцій, у вигляді:

$$dX(t) = (\mu\, X(t) + \xi)dt + (\sigma\, X(t) + \xi)dW(t), \tag{3}$$

де ξ - переключення, які є випадковими та нормально розподіленими величинами на фіксованих інтервалах, що є напівмарковськими процесами.

Напівмарковські процеси

Напівмарковський процес задається співвідношенням[1]:

$$X(t) := X_{v(t)} := X_n, \quad \tau_n \le t < \tau_{n+1}, n \ge 0. \tag{4}$$

Напівмарковський процес є неперервним справа: $X(t+0) = X(t)$. У момент відновлення τ_n, $n \ge 1$ напівмарковський процес змінює свій стан, перебуваючи в кожному стані $X_n = X$ час $\theta_{n+1} = \theta_x$.

Моменти відновлення τ_n, $n \ge 1$ є марковськими моментами для напівмарковського процесу[4]. Це означає, що умовний розподіл напівмарковського процесу не залежить від «минулої історії», якщо фіксовано його стан в момент відновлення τ_n:

$$P\{X(t+\tau_n) \in B / X(\tau_n) = X; X(s), 0 \le s \le t_n\} = P\{X(t) \in B / X(0) = X\} \tag{5}$$

Дифузійний процес

Дифузійні процеси - це неперервні процеси, у яких всюди траєкторії вибірки неперервні, але ніде не диференційовані. Вони виникають у багатьох природних явищах і в якомусь сенсі є узагальненими версіями броунівського руху. Багато функцій можуть бути обчислені в явному вигляді для одновимірного дифузійного процесу[3].

У галузях, де була застосована дифузія, її використовували для моделювання явищ, що змінюються випадково і неперервно за певних умов, наприклад, коливання цін та акцій на ідеальному ринку, варіації зростання населення в ідеальних умовах.

Розв’язок рівняння (1) отримується за допомогою наступного стохастичного інтегрального рівняння[2]:

$$X(t) = X(0) + \int_0^t a(X(u), u)\, du + \int_0^t b(X(u), u)\, dW(u), \quad t \ge 0. \tag{6}$$

Комп’ютерне моделювання дифузійного процесу та опис моделі

Ми розглядаємо процес моделювання цін на акції, який є результатом досліджень, пов'язаних з процесами фінансового моделювання в пошуках зменшення ризиків.

Цей процес описується рівнянням (2), яке має розв’язок:

$$X(t) = X(0)\left\{\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma\, W(t)\right\}, \tag{7}$$

де μ - це параметр переносу, а σ - параметр волатильності, μ і σ - сталі. Згідно з цією моделлю, волатильність залишається постійною протягом декількох періодів, коли відбувається торгівля акціями.

Також ми представляємо таку дифузійну модель, проте з напівмарковськими переключеннями.

Це означає, що період часу розбивається на проміжки θ_i, довжина яких є випадковою і нормально розподіленою величиною. На кожному з цих проміжків моделюємо такий дифузійний процес, де додаємо до волатильності (σ) і коефіцієнту переносу (μ) величину переключення (ξ), яка теж є випадковою і нормально розподіленою величиною.

Тобто на кожному інтервалі θ_i ми розв'язуємо наступне рівняння:

$$dX(t) = (\mu\, X(t) + \xi)dt + (\sigma\, X(t) + \xi)dW(t). \tag{8}$$

Таким чином при кожному новому моделюванні процесу, величини проміжків та величина переключення на певному проміжку будуть змінюватися.

Метою створення програми було надання можливості продемонструвати графічно дифузійний процес та дифузійний процес з марковськими переключеннями.

Вхідними даними є:

X(0) - початкова ціна акції, за замовчуванням 100;

μ – параметр переносу (дрейф), у нашому випадку = 0.2;

σ – параметр волатильності, у нашому випадку = 0.65;

ΔT – період часу для якого розраховуються майбутні ціни, за замовчуванням 52;

dt – деталізація періоду часу (необов’язково).

Отримано масив із змодельованими цінами на акції за даний період часу та графічне зображення.

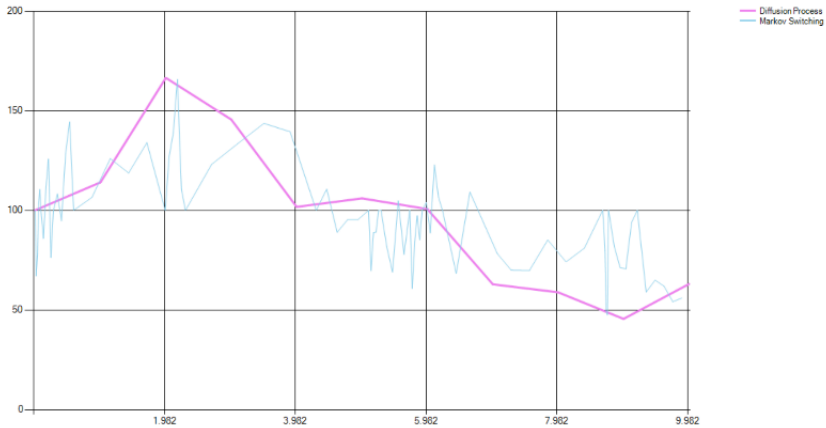


Рис.1 Дифузійний процес з напівмарковськими переключеннями

Висновки

У даній роботі розглянуто дифузійний процес з напівмарковським процесом переключення. Наведено стохастичне диференціальне рівняння дифузійного процесу (1) і (2) та стохастичне диференціальне рівняння дифузійного процесу з напівмарковськими переключеннями виду (3) і їхній розв'язок.

Уперше досліджено модель дифузійного процесу з напівмарковськими переключеннями для цін на акції як розв’язок диференціального стохастичного рівняння.

Наведено геометричну ілюстрацію вихідного дифузійного процесу (2) та модифікованого дифузійного процесу з напівмарковськими процесами (8).

Список літератури

1. Королюк В.С. Стохастичні моделі систем: Навч. посібник. – К.: Либідь, 1993. – 136 с.
2. Скороход А. В. Лекції з теорії випадкових процесів: Навч. посібник. – К.: Либідь, 1990. – 168 с.
3. Oliver C. Ibe Markov Processes for Stochastic Modeling. Second Edition. – Elsevier, 2003. – 514 p.
4. Yaroslav Chabanyuk, Anatolii Nikitin, Uliana Khimka. Asymptotic Analyses for Complex Evolutionary Systems with Markov and Semi-Markov Switching Using Approximation Schemes. – Mathematics and Statistics. John Wiley & Sons. – 2020. – 240 p.

ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ ІГРОВОЇ ПРОГРАМИ PEG SOLITAIRE

Анастасія Петришин

Львівський національний університет ім.Івана Франка

Факультет прикладної математики та інформатики
Anastasiia.Petryshyn@lnu.edu.ua

В процесі розвитку мислення, формуванню особистості, та вмінню швидко і правильно приймати рішення важливу роль відіграє не тільки освітній фактор, але й характер ігрової діяльності, логічні задачі та головоломки. Головоломки зайняли своє місце в наших серцях і головах набагато раніше, ніж будь-які інші іграшки. Всім відомий кубик Рубіка, японські кросворди, п'ятнашки, судоку, різноманітні шаради, пазли, загадки, те що ніколи не набридає людям з давніх давен. Проте зараз, у технічну еру, ці загадки також набувають іншого формату, більш легкого і доступного для всіх людей, у тому що кожен завжди носить у кишені, де кожен може зайти в плеймаркет і завантажити нову головоломку, щоб було чим зайнятись очікуючи когось, або ж у довгій дорозі, або ж просто відволікти мозок від звичної роботи. Розв'язування головоломок — цікавий та захоплюючий процес який здавна затягує мільйони людей по всіх куточках світу. Але грати на телефоні головоломку це одне, а розробити її зовсім інше, набагато важче, та не менш захоплююче завдання.

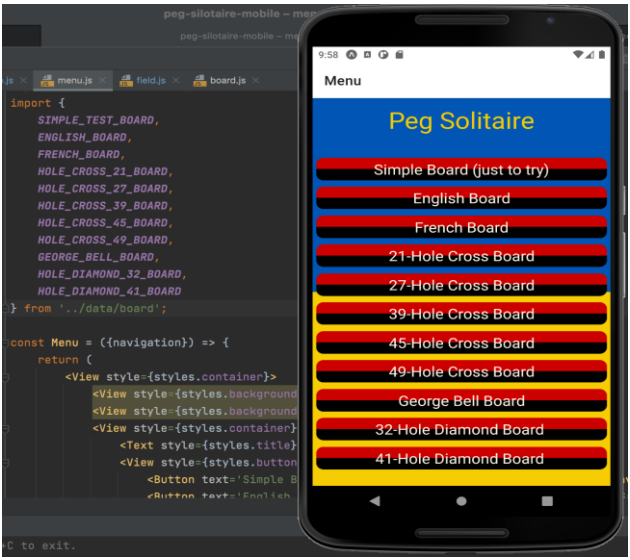
Китайські шашки, Солітер , Peg Solitaire, Patience, Brainvita, Йога — це все назви однієї і тієї самої популярної, насправді Французької, гри-головоломки. Це настільна гра для одного гравця, який має змогу переставляти фішки на дошці з виїмками. У стандартній грі все поле заповнюється фішками, за винятком центрального отвору. Мета гри — звільнити всю дошку від фішок, залишивши останню фішку в центрі дошки.

Для написання гри розглядались такі технології для кросплатформної мобільної розробки як React Native та Flutter. React Native (також відомий як RN) - це популярна платформа мобільних програм на основі JavaScript, яка дозволяє створювати мобільні програми з власним інтерфейсом. React Native, створений і підтримуваний Facebook, є доступною кросплатформною платформою для розробки додатків, яка швидко стала улюбленою серед мобільних розробників. React Native сприяє розробці мобільних додатків для Android та iOS. Найбільш взірцевими прикладами програм React Native є

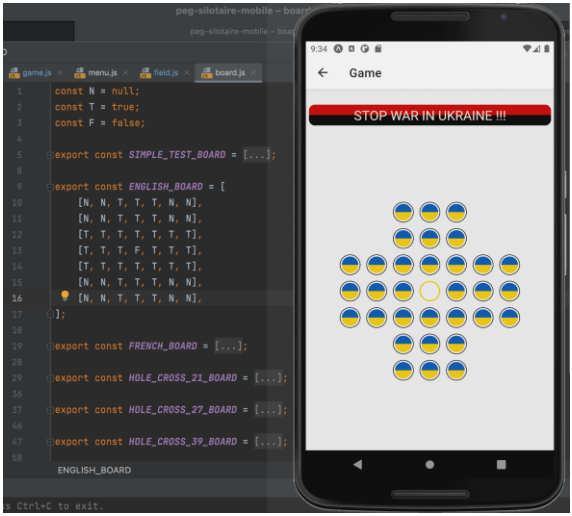
програми відомих компаній, таких як Tesla, Airbnb, Skype або Amazon Prime. Головна привабливість React Native полягає в тому, що він забезпечує швидшу розробку та легку підтримку. Повторно використовувані компоненти, можливість використовувати величезної кількості наявних JS бібліотек з NPM, а також створення графічного інтерфейсу на основі CSS подібного синтаксису є привабливими аргументами для вибору React Native.

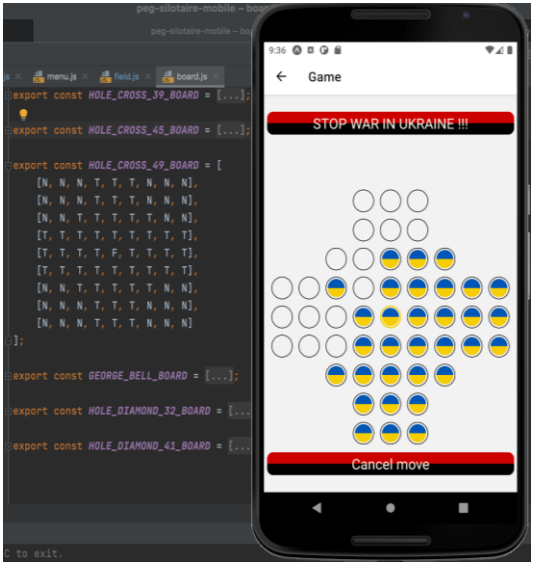
Flutter — це відкритий і безкоштовний фреймворк від Google, який дозволяє створювати програми для Android та iOS за допомогою простої кодової бази. Це інноваційний пакет SDK для розробки міжплатформених додатків. Flutter всеосяжний мобільний фреймворк, який містить віджети, механізм візуалізації та налагодження, а також ресурси для допомоги розробникам у створенні та розгортанні красивих мобільних додатків. Flutter використовується рядом відомих організацій, включаючи Google і Abbey Road Studios.

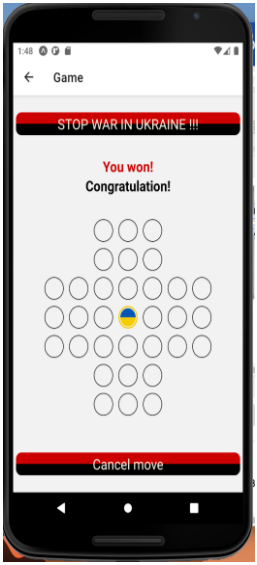
Для розробки додатку була використана платформа React Native, оскільки вона доволі проста у використанні, найбільш популярна на сьогоднішній день, має багато документації та прикладів. Такі мобільні додатики як Facebook, Instagram та Uber були розроблені з використанням цієї платформи. Дані застосунки без проблем запускаються на будь якому пристрої, а для ігор це доволі важливо.



У гру було додано 11 рівнів, для різних варіантів ігрової дошки. Один рівень задається матрицею, який містить саму мапу рівня (двовимірний масив, який складається з кодових символів для пізнішого накладання зображень). За даними, прописаними у мапі рівня, створюється грід, у який передаються кодові символи для правильного зображення та відображення гри.







Для здійснення ходу спочатку гравець натискає на фішку, яку бажає переставити. Пізніше на місце, куди збирається переставити. Далі відбувається перевірка чи це можливо (фішки вибрані правильно, фішка переставляється в місце, де немає іншої фішки, переставляється лише по горизонталі або по вертикалі, лише на 2 кроки та між цими місцями є фішка, яка має перескакуватись і забиратись). Тільки при виконанні всіх умов хід відбудеться. Для початку у списку зберігається поточний стан гри (це потрібно для реалізації функції ‘назад’). Далі обрана фішка переміститься в задане місце а фішка між цими місцями забирається з поля. Ходи відбуваються доти, доки на полі не залишиться 1 фішка.

У даній роботі було реалізовано повноцінний мобільний застосунок, використовуючи технологію React Native. Розроблено усі основні функції гри, 10 повноцінних рівнів та 1 тестовий. Використовуючи React Native, можна швидко, якісно і просто реалізувати клосплатформлений мобільний додаток гри і який можна легко підтримувати. React Native дозволяє ефективно стилізувати аплікацію з адаптацією під різні розміри дисплеїв.

Код реалізації мобільного додатку гри знаходиться на GitHub за посиланням https://github.com/Stacy1256/peg-silotaire-mobile.

Список використаних джерел

1. React native (documentation) - https://reactnative.dev/
2. Peg Solitaire (history and rules)- https://sites.google.com/site/boardandpieces/list-of-games/peg-solitaire
3. Advantages and Disadvantages of React Native Development - https://www.thirdrocktechkno.com/blog/pros-and-cons-of-react-native-development-in-2021/

КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ СЦЕН З УРАХУВАННЯМ ЗАЛОМЛЕННЯ ПРОМЕНІВ СВІТЛА

Софія Ковальчук

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

Sofiia.Kovalchuk@lnu.edu.ua

Вступ

На сьогодні комп'ютерне моделювання вважається однією з провідних галузей, яка використовується авіа-, машинобудівними фірмами під час конструювання та виготовлення виробів, які потребують високої точності, а основною задачею комп'ютерного моделювання є візуалізація, тобто створення зображення на основі опису (моделі) того, що потрібно зобразити. Поступово ця галузь увійшла у різні сфери повсякденного життя. До основних сфер застосування її технологій можна віднести такі:

– графічний інтерфейс користувача;

– спецефекти, візуальні ефекти, цифрова кінематографія;

– цифрове телебачення, відеоконференції;

– цифрова фотографія;

– цифровий живопис;

– візуалізація наукових та ділових даних;

–комп'ютерні ігри, системи віртуальної реальності (наприклад, тренажери з керування автомобілем, літаком тощо);

– системи автоматизованого проектування;

– комп'ютерна томографія.

Цей перелік постійно розширюється, бо використання комп'ютерного моделювання у різних сферах знань дає поштовх новим напрямкам.

Моделювання графічних об'єктів використовує математичні методи, досягнення у галузях фізики, природничих наук. Для реалістичного зображення сцен необхідно враховувати як геометрію складових об'єктів сцени, так і фізичні властивості матеріалів, з яких виготовлені об'єкти, біологічну будову живих об'єктів тощо.

Очевидно, що середовище, яке нас оточує, ми бачимо завдяки тому, що всі предмети відбивають світло, пропускають його через себе, випромінюють. Для передачі та збереження кольору в комп'ютерній графіці використовуються різні форми його представлення. У загальному випадку

колір – набір чисел, які представляють собою координати у певній колірній моделі.

Для моделювання тривимірних об’єктів необхідно враховувати закони поширення світла у середовищах, властивості об’єктів відбивати, заломлювати, випромінювати світло.

У своїй роботі я займалась дослідженнями, які пов’язані з принципами побудови та з використанням законів поширення світла для моделювання просторових сцен.

Постановка задачі

Для повноцінного дослідження даної теми я виділила для себе декілька етапів:

1. дослідження існуючих методів та засобів відтворення та підвищення реалістичності відтворення сцен у засобах комп’ютерної графіки.
2. дослідження фізичної складової заломлення променів світла та існуючих аналогів математичних моделей для побудови різних видів сцен у комп’ютерному моделюванні.
3. розробка програмного модуля для відтворення сцен з урахуванням заломлення променів світла у комп’ютерному моделюванні, які потенційно зменшують час формування сцен.

В ході першого етапу досліджень я зробила висновок, що розрахунок освітлення є алгоритмічно складною задачею. Навіть з урахуванням усіх спрощень та компромісів, що допускаються у ряді методів моделювання освітлення, результуюча складність усе одно може бути завеликою і затрати часу на візуалізацію окремого кадру можуть перевищувати визначений максимальний ліміт.

Відповідно до цього було досліджено існуючі аналоги математичних моделей для побудови різних видів сцен та розроблено певні додаткові оптимізації щодо них для можливості зменшення використання часу на процес візуалізації, а також спроби використання більш складних та якісних методів моделювання освітлення, щоб, за рахунок цього, за загальним часом візуалізації залишитись у встановлених вимогами межах та зі збереженням якості сцен.

Висновки

В ході дослідження та опрацювання даної теми мною було створено програмний модуль з використанням алгоритму на основі методу трасування променів для збереження якості сцен з урахуванням заломлення променів

світла відповідно до заданих матеріалів при зменшенні часу на обробку та формування даної сцени, а також ваги вихідного файлу та мови програмування Python завдяки гнучкості та простоті синтаксису даної мови.

Саме метод трасування променів увійшов в основу за рахунок можливості рендерингу гладеньких об'єктів без апроксимації їх полігональними поверхнями, відсічення невидимих поверхонь, перспективи і коректних змін поля зору, які є логічним наслідком алгоритму та низки переваг над іншими методами, які були розглянуті в ході досліджень, за рахунок яких було отримано зменшення часу на генерацію одного кадру результуючої сцени з мінімальними втратами по якості для користувача.

На даний момент основним призначенням даного модулю є його використання, як бібліотеки для використання різними програмами, які створені як системи віртуальної реальності, так і для професійних цілей, зокрема перетворювачів для зменшення розмірів вхідних файлів для полальшого опрацювання.

Список літератури

1. Лисиця В.Т. Колірні моделі та закони поширення світла / В. Т. Лисиця. – Х. : ХНУ імені В.Н. Каразіна, 2012. – 82 с
2. Шикін Є. В. Комп'ютерна графіка. Полігональні моделі (рос.) / Є. В. Шикін, О. В. Борєсков. – М. : Діалог МІФІ,2001. – 462 с
3. Han-Wei Shen. Ray tracing basics [Електронний ресурс]. – Режим доступу: http://web.cse.ohiostate.edu/~hwshen/681/Site/Slidesfiles/basicalgo.pdf (дата звернення: 19.10.2021).
4. Foley J.D., van Dam A. Feiner S.K., Hughes J.F. Computer Graphics: Principles and Practice // Addison-Wesley Publishing Company. – 1996. – P. 718–739.
5. Phong B.T. Illumination for computer generated pictures // Communications of ACM. 1975. – № 6. – P. 311–317.

ПЕРЕДБАЧЕННЯ НАГРОМАДЖЕННЯ ТРАФІКУ ЗА ДОПОМОГОЮ СИМУЛЯЦІЇ ТА ШТУЧНОГО ІНТЕЛЕКТУ

Олег Іванків, Микола Ковалик

Дублянський опорний ліцей імені Героя України Анатолія Жаловаги; Національний університет "Львівська політехніка"

oleguk0509@gmail.com, niciceberg.public@gmail.com

Вступ

Дорожня інфраструктура є артерією міст, чим злагодженіше вона працює, тим швидше розвивається місто. Для прийняття відповідних заходів, урбаністи та мерії міст потребують все більш детальних даних. Такі заходи, як, наприклад, встановлення обмеження на швидкість на певній дорозі вимагають даних про ситуацію нагромадження трафіку у даній місцевості. Збір таких даних є дуже дорогою процедурою, оскільки треба проводити постійне спостереження багатьма датчиками. Обмеженість бюджету міських адміністрацій не дозволяє виконувати такі процедури, а отже, прийняті заходи зазвичай виявляються неефективними.

Створення нового ефективного та дешевого методу, що дозволить здобувати інформацію про стан дорожньої інфраструктури міста дозволить значно скоротити витрати та підвищити якість впроваджених заходів.

Наявні методи передбачення нагромадження трафіку

Методів передбачення трафіку є досить багато, однак практичні усі з них застосовують темпоральні дані, тобто передбачення, які вони створюють, є виключно тимчасовими, тож їх ефективність визначається у тому, наскільки довго вони можуть точно передбачати нагромадження.

Вирізняють три групи методів передбачення трафіку [1]:

- Методи ймовірностей. Дані методи досить точно визначають характер нагромадження на окремих дорогах, але не здатні обробляти масив даних усієї дорожньої інфраструктури.
- Неглибоке машинне навчання. Ці методи мають значний недолік, це те, що вони не можуть самі визначати особливості об'єктів, що робить процес їх навчання досить складним процесом.
- Глибоке машинне навчання. На відміну від неглибокого машинного навчання, вони здатні автоматично визначати особливості об'єктів. Дана категорія методів є найкращою для розв'язання поставлених задач у даній

науковій роботі, оскільки ручне визначення характеристик дорожньої мережі є практично нереальною задачею.

Метод симуляції

Для будь-якого існуючого методу передбачення трафіку потрібно великі обсяги інформації. Для отримання результатів у різних, подібних до реальних, ситуаціях, найкраще підійде метод симуляції. Метод складається з двох основних частин: генерація дорожньої мапи та поведінка автомобілів. Дорога генерується за допомогою кривих Безьє та працює за моделлю теорії графів. Своєю чергою, машини працюють за принципами симулювання фізичних об’єктів та оснащені алгоритмами логіки вирішення ситуацій, як наприклад поведінка на перехресті.

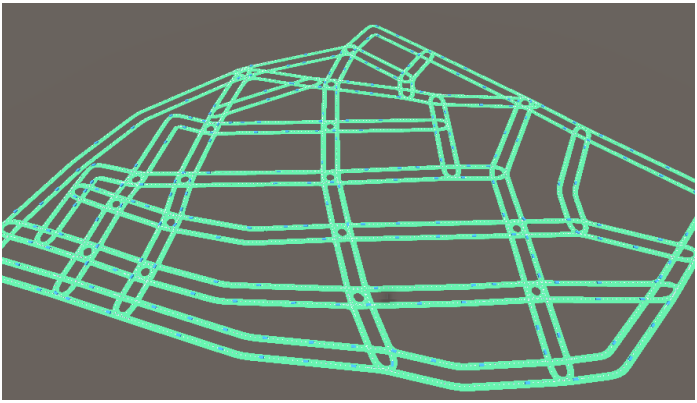


Рис. 1. Згенерована дорожня мапа

Для збору даних симуляції, генеруються сектори слідкування, у яких є два важливих параметри - це occupancyTime (час коли у секторі було авто) та occupancyRate, відношення часу симуляції до occupancyTime. Дані експортуються шляхом створенням мапи значень occupancyRate кожної секції дорожньої інфраструктури.

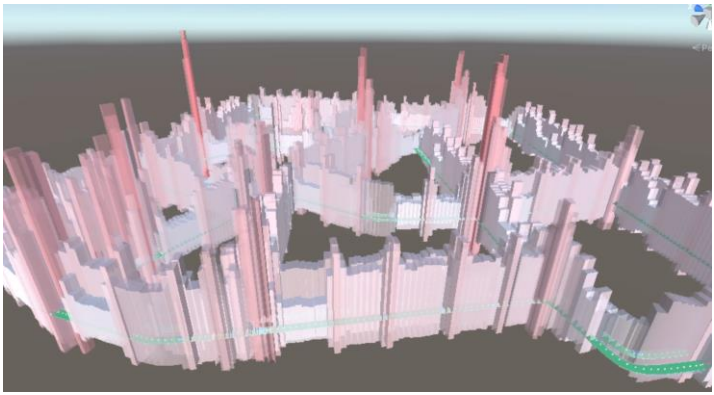


Рис. 2. Сектори зчитування даних фрагменту дороги

Метод штучного інтелекту

Штучний інтелект побудований на принципі пошарової нейронної мережі із навчання з вчителем. Першим шаром є згортковий шар з 10 фільтрами, переважно саме він використовується для знаходження в мапах закономірностей, дальше ідуть 3 приховані шари, що використовуються RELU функції, та один шар з вихідними даними, який використовує Sigmoid функцію. Для тренування використовується лінійна cost-function, а для запобігання запам'ятовування передбачених патернів використовуються Drop-out шари. Важливо зазначити, що перед тим, як дані вводяться у нейронну мережу, вони сортуються, зменшуються (розширення мапи) та нормалізуються (зведення даних до бінарних значень).

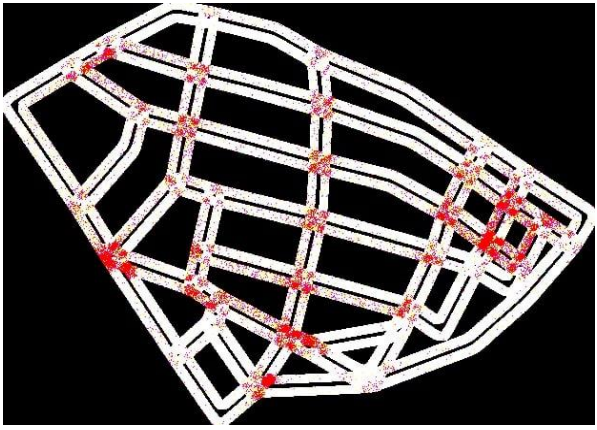


Рис. 3. Вихідне зображення тренованої нейромережі

Висновки

В ході роботи було створено два методи передбачення нагромадження трафіку, проаналізовано фактори, що можуть на нього впливати та проведено дослідження з перевірки ефективності даних методів. Розроблені методи використовують нетемпоральні дані, а отже, є незалежними від будь-яких інших факторів, що робить їх зручними для використання у подальших аналізах. Ефективність даних методів є досить висока, так наприклад, дані, які виводить штучний інтелект мають точність 90%. Створено алгоритм зчитування та експортування даних симуляції, систему поведінки автомобілів, генерацію структури та об’єктів дорожньої інфраструктури.

Список літератури

1. M. Akhtar, A Review of Traffic Congestion Prediction Using Artificial Intelligence / M. Akhtar, S. Moridpour // Journal of Advanced Transportation, - 2021, 18 с.
2. X. Yang, Application of Data Science Technologies in Intelligent Prediction of Traffic Congestion / X. Yang, S. Luo, K. Gao, T. Qiao, X. Chen // Journal of Advanced Transportation, - 2019, 18 с.
3. N. Kumar, Applications of deep learning in congestion detection, prediction and alleviation A survey / N. Kumar, M. Raubal // Transportation Research, - 2021, 27 с.
4. Z. Guo, Urban traffic modeling and pattern detection using online map vendors and self-organizing maps / Z. Guo, B. Li, L. Hovestadt // Higher Education Press, -2021, 14с.
5. Alberto Bull Compilador. Traffic congestion the problem and how to deal with it, Сантьяго, 2003 - 198 с.

ПОБУДОВА РОЗПОДІЛЕНОЇ СИСТЕМИ ДЛЯ РЕПЛІКАЦІЇ ДАНИХ

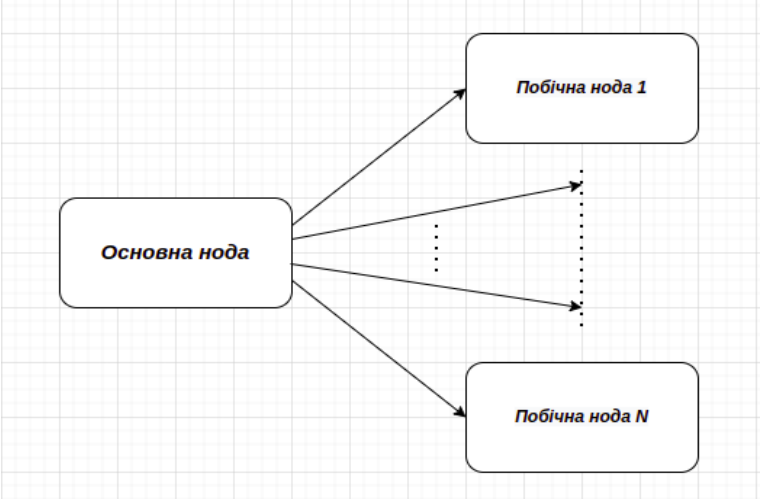
Богдан-Микола Добосевич

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

bohdan.dobosevych@gmail.com

Багато IT компаній зараз займається реплікацією своїх даних для того, щоб спростити доступ і роботу з ними. Одне з рішень - це використання хмарних сховищ, але не завжди це є фінансово вигідно, оскільки вони досить дорогі. Реплікувати дані на ту саму машину, де зберігаються основні дані не пришвидшує доступ до них. Альтернативою є побудова розподіленої системи.



Малюнок 1. Приклад розподіленої системи

У даній роботі будували розподілену систему в форматі один до багатьох, це система, яка передбачає основну ноду і кількох побічних нод. Основна нода буде отримувати дані, які вона буде асинхронно відправляти на побічні ноди.

При побудові такої системи слід враховувати:

1. Система має легко масштабуватися.
2. Система має передбачати роботу з незалежними користувачами.
3. Закритість системи, побічні ноди мають приймати нові дані лише з основної нада.
4. Система має бути стійкою до збоїв.

5. Аналізувати ноди в системі.

Щодо першого пункту, для легкого масштабування використовували docker і docker compose. Docker — інструментарій для управління ізольованими Linux-контейнерами.

Щодо другого пункту, система реалізована, так що користувачі є незалежними між собою і вони не впливатимуть на роботу один одного.

Щодо третього пункту, кожна нода - це gRPC сервер і між ними відбувається спілкування. gRPC — це сучасна високопродуктивна платформа віддаленого виклику процедур (RPC), з відкритим кодом, розроблена в Google у 2015 році.

Щодо четвертого пункту, система є стійка до збоїв. Користувачі можуть нею користуватися при відсутності зв'язку з кількома побічними нодами. Також система буде до надсилати дані при відновленні зв'язку чи додавання нових побічних нод.

Щодо п'ятого пункту, в система використовує харбіти і кворум.

Харбіти - це запит який будуть відправлятися з основної нода на побічні ноди кожну секунду. Аналізувавши за який час ми отримуємо відповідь ми скажемо в якому з трьох станів знаходиться нода.

Кворум - це правило за яким користувач з може працювати з системою.

У даній роботі я розглянув основні переваги і недоліки побудови розподіленої системи для реплікації даних. На практиці довелося працювати з новими технологіями з якими я не працював під час навчання.

Література

1. Distributed computing [Електронний ресурс]. - Режим доступу: https://en.wikipedia.org/wiki/Distributed_computing
2. Replication (computing) [Електронний ресурс]. - Режим доступу:https://en.wikipedia.org/wiki/Replication_(computing)
3. Quick start gRPC Посібник для роботу з у Python. [Електронний ресурс]. - Режим доступу:https://grpc.io/docs/languages/python/quickstart/
4. Docker [Електронний ресурс]. - Режим доступу: https://www.docker.com/
5. Docker Compose [Електронний ресурс]. - Режим доступу: https://docs.docker.com/compose/
6. Thread-based parallelism [Електронний ресурс]. - Режим доступу: https://docs.python.org/3/library/threading.html
7. CountDownLatch in Python using Condition Objects [Електронний ресурс]. - Режим доступу:https://www.madhur.co.in/blog/2015/11/02/countdownlatch-python.html

РОЗПІЗНАВАННЯ ПРЕДМЕТІВ ОДЯГУ З ВИКОРИСТАННЯМ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ

Софія Яропуд

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

Sofiia.Yaropud@lnu.edu.ua

У зв'язку з масштабним розвитком інтернет-застосунків, актуальність проблеми розпізнавання реальних об'єктів з зображення є дуже великою.

Для таких задач найчастіше використовують механізм штучних нейронних мереж [2].

У роботі використовується цей підхід для окремої прикладної задачі розпізнавання та визначення реальних розмірів предметів одягу.

Задача включає попередню обробку вхідного зображення, розпізнавання, класифікацію предметів одягу, сегментацію окремих складових, визначення розмірів характерних складових предметів одягу.

У роботі застосовано теорію розпізнавання на основі нейронних мереж з навчанням моделі MNIST TensorFlow [3], збереження навченої мережі та її використання для класифікації зображень.

Сформулюємо проблему: користувач отримує фотографію з зображенням певного предмета одягу з еталонним елементом.

Кінцева мета дослідження: отримати класифікацію предмета одягу і відповідно до цієї класифікації визначити характерні розміри деталей, відштовхуючись від розмірів еталонного елемента.

На першому етапі слід підготувати зображення для розгляду: очистити його від шумів та здійснити процес контрастування. Цей етап не є предметом дослідження даної роботи. Отже, вважатимемо, що зображення є опрацьоване.

На другому етапі реалізується процес машинного навчання за допомогою відомої моделі MNIST з бібліотеки TensorFlow. Слід зазначити, що для такої задачі система MNIST містить базу даних зображень для тренування навчальної моделі Fashion MNIST, яка містить 70 000 зображень у відтінках сірого у 10 категоріях. Зображення показують окремі предмети одягу з низькою роздільною здатністю (28 на 28 пікселів). Дані зображення не містять еталонних елементів. 60 000 зображень використовуються для навчання мережі та 10 000 зображень для оцінки того, наскільки точно мережа навчилася класифікувати зображення.

Для моделі встановлюємо функцію втрати, оптимізатор, метрики та здійснюємо навчання мережі. Навчена система зберігається і використовується для верифікації користувацьких зображень, які спочатку форматуємо до потрібного вигляду. Для цього змінюємо розмір зображення та здійснюємо перетворення в сіру колірну гаму. Класифікуємо зображення, використовуючи бібліотечні функції [5], та здійснимо для кожного класу блок дій визначення розмірів потрібних частин одягу. Алгоритм знаходження розмірів для штанів був описаний в роботі [4]. Використавши ідейно схожий підхід, розроблено алгоритм обчислення характеристик для інших предметів одягу.

Для реалізації цього алгоритму створений програмний застосунок на мові Python [1]. Продемонструємо отриманий результат для двох предметів одягу: штанів та футболки. На рис.1,2 подано вхідні зображення. На рис. 3,4 подано результат розпізнавання. Як бачимо, система правильно ідентифікувала предмет одягу з еталонним елементом. На рис. 5,6 відображено результат пошуку характерних особливостей визначеного класу та визначення їх розмірів.

Рис. 1. Вхідне зображення штанів.

Рис. 2. Вхідне зображення футболки.

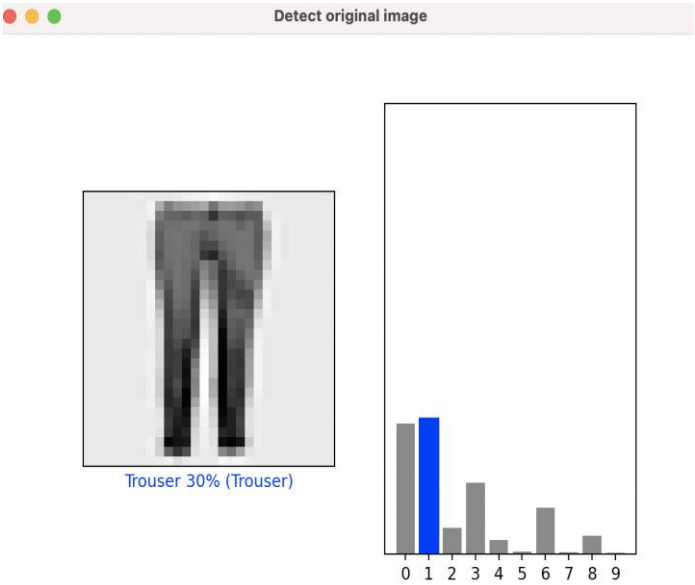


Рис. 3. Результат розпізнавання штанів.

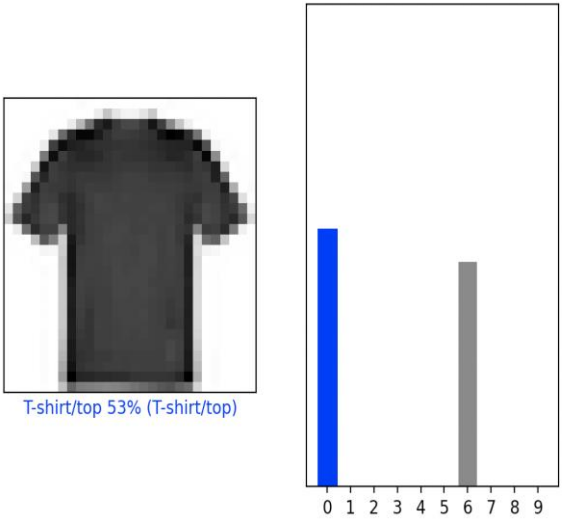


Рис. 4. Результат розпізнавання футболки.

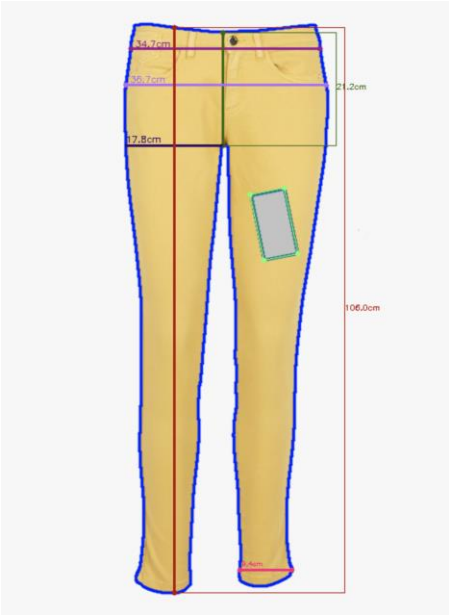


Рис. 5. Результат визначення характерних розмірів штанів.

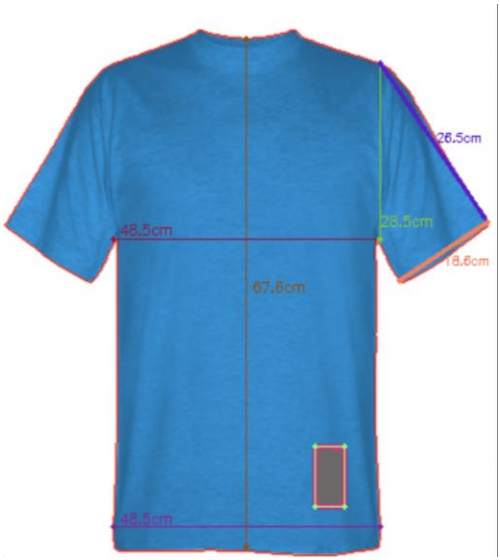


Рис. 6. Результат визначення характерних розмірів футболки.

Чисельні експерименти показали, що отримана модель добре працює для відносно стандартних фасонів елементів одягу з однотонних тканин. Розв’язування проблеми без цих обмежень вимагає додаткових досліджень.

Список використаних джерел:

1. Adrian Rosebrock. Practical Python and OpenCV [Електронний ресурс] – Режим доступу до ресурсу: https://minhtn1.github.io/Practical%20Python%20and%20OpenCV,%203rd%20Edition.pdf
2. Charu C. Aggarwal. Neural Networks and Deep Learning. - Springer. - 2018. - 520 p.
3. TensorFlow documentation. Classify images of clothing [Електронний ресурс] – Режим доступу до ресурсу: https://www.tensorflow.org/tutorials/keras/classification.
4. Яропуд С. Визначення точних розмірів предметів одягу з використанням OpenCV / Тези доповідей міжнародної студентської наукової конференції з питань прикладної математики та комп’ютерних наук МСНКПМК-2021. - 2021. - С. 8-10.
5. Alexey Spizhevoy. OpenCV 3 Computer Vision with Python / Alexey Spizhevoy, Aleksandr Rybnikov. - Packt Publishing. - 2018. - 306 p.

INFORMATION RETRIEVAL SYSTEM DEVELOPMENT 4

РОЗРОБКА СИТЕМИ КЛАСТЕРИЗАЦІЇ АНТИТІЛ НА ОСНОВІ КОЕФІЦІЄНТУ ПЕРЕХРЕСНОГО ЗВ’ЯЗУВАННЯ..... 8

ДОСЛІДЖЕННЯ ВИКОРИСТАННЯ РІЗНИХ ДВОВИМІРНИХ ТРИКУТНИХ ЕЛЕМЕНТІВ ДЛЯ МОДЕЛЮВАННЯ СЕРЕДОВИЩ МЕТОДОМ СКІНЧЕННИХ ЕЛЕМЕНТІВ 13

АВТОРИЗАЦІЯ КОРИСТУВАЧА ЗА ГОЛОСОМ 17

ЧИСЛОВЕ МОДЕЛЮВАННЯ МЕТОДОМ СКІНЧЕННИХ ЕЛЕМНТІВ ЗАДАЧ ТЕПЛОПРОВІДНОСТІ..... 21

ВИКОРИСТАННЯ ГЕНЕТИЧНИХ АЛГОРИТМІВ ДЛЯ ЧИСЕЛЬНОГО РОЗВ'ЯЗУВАННЯ ДВОВИМІРНИХ ІНТЕГРАЛЬНИХ РІВНЯНЬ ПЕРШОГО РОДУ 25

ЧИСЕЛЬНЕ РОЗВ'ЯЗУВАННЯ ОБЕРНЕНОЇ ЗАДАЧІ ТЕПЛОПРОВІДНОСТІ ПАРАЛЕЛЬНИМИ ГЕНЕТИЧНИМИ АЛГОРИТМАМИ 29

INVESTIGATION AND IMPLEMENTATION OF THE ITEM-BASED COLLABORATIVE FILTERING 33

ПРОГРАМНО-АЛГОРИТМІЧНЕ ЗАБЕЗПЕЧЕННЯ ОПТИЧНОГО РОЗПІЗНАВАННЯ ТЕКСТУ ЗОБРАЖЕНЬ НИЗЬКОЇ ЯКОСТІ МЕТОДАМИ МАШИННОГО НАВЧАННЯ .. 36

ЗАСТОСУВАННЯ ПАРАЛЕЛЬНИХ СХЕМ ПРИ РОЗВ’ЯЗУВАННІ СИСТЕМ МАТРИЧНИХ РІВНЯНЬ..... 41

ПРОЕКТУВАННЯ ТА РОЗРОБКА МОБІЛЬНОГО АСИСТЕНТА "WINGMAN" 45

РОЗПІЗНАВАННЯ ОБ’ЄКТІВ У РЕАЛЬНОМУ ЧАСІ ЗА ДОПОМОГОЮ TENSORFLOW У МОБІЛЬНИХ ДОДАТКАХ 47

MATHEMATICAL MODELING OF ACOUSTIC WAVE PROPAGATION IN VISCOUS FLUID..... 50

MATHEMATICAL MODELING AND ANALYSIS OF STATIONARY ACOUSTIC WAVE PROPAGATION IN VISCOUS FLUID 54

РОЗРОБКА ЧАТ-БОТА МОВОЮ ПРОГРАМУВАННЯ PYTHON 59

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ТЕСТУВАННЯ ПРОДУКТИВНОСТІ ПЕРСОНАЛЬНИХ КОМП’ЮТЕРІВ 63

ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ ЗАРЯДНОЇ СМАРТ-СТАНЦІЇ AKER 67

OBJECT DETECTION AND SIZE RECOGNITION 71

ВИЯВЛЕННЯ ТА РОЗПІЗНАВАННЯ ОБ’ЄКТІВ ЗА ДОПОМОГОЮ АЛГОРИТМІВ МАШИННОГО НАВЧАННЯ ОНЛАЙН..... 75

ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПРОГНОЗУВАННЯ ДИНАМІКИ КУРСІВ КРИПТОВАЛЮТ ІЗ ВИКОРИСТАННЯМ ФРАКТАЛЬНОГО АНАЛІЗУ 79

МОДЕЛЬ МАШИННОГО НАВЧАННЯ ДЛЯ РОЗПІЗНАВАННЯ АНГЛІЙСЬКОГО ДАКТИЛЬНОГО АЛФАВІТУ..... 83

ПОРІВНЯННЯ ЕФЕКТИВНОСТІ ВИКОРИСТАННЯ РЕЛЯЦІЙНОЇ МОДЕЛІ ДАНИХ ТА МЕРЕЖЕВОЇ МОДЕЛІ ДАНИХ У ВИГЛЯДІ ГРАФУ ДО ТІСНО ПОВ'ЯЗАНИХ ДАНИХ НА ПРИКЛАДІ СОЦІАЛЬНОЇ МЕРЕЖІ 85

.NET-БІБЛІОТЕКА МАТЕМАТИЧНИХ ЗАСТОСУНКІВ APPLIEDMATHLIBRARY 88

МАТЕМАТИЧНА МОДЕЛЬ ПОШИРЕННЯ COVID-19 90

РЕАЛІЗАЦІЯ РІЗНОМАСШТАБНОГО МЕТОДУ СКІНЧЕННИХ ЕЛЕМЕНТІВ ДЛЯ ДВОВИМІРНОЇ ЛІНІЙНОЇ КРАЙОВОЇ ЗАДАЧІ 95

IMPROVE THE FREQUENCY RESOLUTION OF A 3D 99

PHYSICALLY BASED SIMULATION OF MAJOR EARTHQUAKE 99

МЕТОДИ ТИПУ НЬЮТОНА ДЛЯ РОЗВ'ЯЗУВАННЯ ОБЕРНЕНОЇ ЗАДАЧІ НА ВЛАСНІ ЗНАЧЕННЯ 102

ПРОТОТИП САЙТУ НА ОСНОВІ СТАТИЧНИХ ДОСЛІДЖЕНЬ ТА ВИБОРУ ОПТИМАЛЬНОЇ СТРАТЕГІЇ 107

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ НАДІЙНОГО ТА ЕФЕКТИВНОГО ШИФРУВАННЯ ДАНИХ 110

MICROSCOPIC SIMULATION OF ROAD TRAFFIC 112

DETECTING PARKING OCCUPANCY USING MACHINE LEARNING 117

МОДЕЛЬ ДИФУЗІЙНОГО ПРОЦЕСУ З НАПІВМАРКОВСЬКИМИ ПЕРЕКЛЮЧЕННЯМИ 124

ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ ІГРОВОЇ ПРОГРАМИ PEG SOLITAIRE 128

КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ СЦЕН З УРАХУВАННЯМ ЗАЛОМЛЕННЯ ПРОМЕНІВ СВІТЛА 132

ПЕРЕДБАЧЕННЯ НАГРОМАДЖЕННЯ ТРАФІКУ ЗА ДОПОМОГОЮ СИМУЛЯЦІЇ ТА ШТУЧНОГО ІНТЕЛЕКТУ 135

ПОБУДОВА РОЗПОДІЛЕНОЇ СИСТЕМИ ДЛЯ РЕПЛІКАЦІЇ ДАНИХ 139

РОЗПІЗНАВАННЯ ПРЕДМЕТІВ ОДЯГУ З ВИКОРИСТАННЯМ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ 141