

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені Ігоря СІКОРСЬКОГО»
НАВЧАЛЬНО-НАУКОВИЙ ФІЗИКО-ТЕХНІЧНИЙ
ІНСТИТУТ

Кафедра математичного моделювання та аналізу даних

«До захисту допущено»

В.о. завідувача кафедри

_____ І.М. Терещенко

«___» _____ 2024 р.

Дипломна робота
на здобуття ступеня бакалавра

зі спеціальності: 113 Прикладна математика
на тему: «Порівняння багатосарового перцептрону та $(1 + \lambda)$ -
еволюційного алгоритму для задач класифікації»

Виконав: студент 4 курсу, групи ФІ-02
Харь Дмитро Федорович

Керівник: асистент кафедри ММАД Яворський О.А. _____

Рецензент: звання, степінь, посада Прізвище І.П. _____

Засвідчую, що у цій дипломній
роботі немає запозичень з праць
інших авторів без відповідних
посилань.

Студент _____

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені Ігоря СІКОРСЬКОГО»
НАВЧАЛЬНО-НАУКОВИЙ ФІЗИКО-ТЕХНІЧНИЙ
ІНСТИТУТ

Кафедра математичного моделювання та аналізу даних

Рівень вищої освіти — перший (бакалаврський)
Спеціальність (освітня програма) — 113 Прикладна математика,
ОПП «Математичні методи моделювання, розпізнавання образів та
комп'ютерного зору»

ЗАТВЕРДЖУЮ

В.о. завідувача кафедри

_____ І.М. Терещенко

«__» _____ 2024 р.

ЗАВДАННЯ
на дипломну роботу

Студент: Харь Дмитро Федорович

1. Тема роботи: *«Порівняння багатошарового перцептронну та $(1 + \lambda)$ -еволюційного алгоритму для задач класифікації»*,

керівник: асистент кафедри ММАД Яворський О.А.,

затверджені наказом по університету №__ від «__» _____ 2024 р.

2. Термін подання студентом роботи: «__» _____ 2024 р.

3. Вихідні дані до роботи:

4. Зміст роботи: *Порівняльний аналіз багатошарового перцептронну (англ. MLP, Multilayer Perceptron) з оптимізаційними алгоритмами в основі яких градієнтний спуск, MLP з оптимізаційним алгоритмом в основі якого одноточкова мутація та $(1 + \lambda)$ -еволюційного алгоритму з кодуванням генетичного програмування (англ. $(1 + \lambda)$ -EA with GP encoding, $(1 + \lambda)$ -evolutionary algorithm with genetic programming encoding), на прикладі задач бінарної та мультикласової класифікації табличних даних та картинок.*

5. Перелік ілюстративного матеріалу: *«Презентація доповіді»*

6. Дата видачі завдання: 10 грудня 2023 р.

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання	Примітка
1	Узгодження теми роботи із науковим керівником	листопад- грудень 2023 р.	Виконано
2	Огляд та опрацювання опублікованих джерел за тематикою дослідження	грудень 2023 р - лютий 2024 р.	Виконано
3	Написання програмного забезпечення та проведення дослідження	березень-квітень 2024 р.	Виконано
4	Оформлення та опис результатів	травень 2024 р.	Виконано
5	Написання та оформлення дипломної роботи	травень-червень 2024 р.	Виконано
6	Отримання рекомендації до захисту	08.06.2024	Виконано

Студент

_____ Харь Д.Ф.

Керівник

_____ Яворський О.А.

РЕФЕРАТ

Кваліфікаційна робота містить: ??? стор., ??? рисунки, ??? таблиць, ??? джерел.

У даній роботі розглядаються методи для вирішення задач класифікації, а саме: MLP, який використовує оптимізаційні алгоритми в основі яких градієнтний спуск, MLP, який використовує оптимізаційний алгоритм на основі одноточкової мутації та $(1 + \lambda)$ -EA with GP encoding. Ці методи порівнювались в задачах бінарної та мультикласової класифікації табличних даних та картинок.

У ході дослідження було встановлено, що всі три методи здатні досягти однакових метрик у всіх задачах. Найшвидшу сходимість до цих метрик продемонстрував MLP з використанням градієнтного спуску. Тим не менш, $(1 + \lambda)$ -EA з генетичним програмуванням виділився завдяки здатності легко адаптуватись до умов задачі. Цей метод дозволяє вибирати кількість нащадків і тип мутацій, що надає можливість зосередити пошук рішень у конкретних областях простору рішень. Такий підхід є особливо корисним, коли потрібно зосередитися на важливих регіонах пошуку для вдосконалення рішень.

МАШИННЕ НАВЧАННЯ, ЕВОЛЮЦІЙНІ АЛГОРИТМИ,
ГЕНЕТИЧНЕ ПРОГРАМУВАННЯ, МЕТОДИ ОПТИМІЗАЦІЇ,
ЕКСПРЕСИВНІ КОДУВАННЯ

ABSTRACT

This paper considers methods for solving classification problems, namely: MLP, which uses optimization algorithms based on gradient descent, MLP, which uses an optimization algorithm based on one-point mutation, and $(1+\lambda)$ -EA with GP encoding. These methods were compared in the tasks of binary and multiclass classification of tabular data and pictures.

During the research, it was established that all three methods are able to achieve the same metrics in all tasks. The fastest convergence to these metrics was demonstrated by MLP using gradient descent. Nevertheless, the $(1+\lambda)$ -EA with genetic programming stood out due to its ability to easily adapt to the task conditions. This method allows you to choose the number of offspring and the type of mutations, which makes it possible to focus the search for solutions in specific regions of the solution space. This approach is particularly useful when focusing on important search regions to improve solutions.

MACHINE LEARNING, EVOLUTIONARY ALGORITHMS,
GENETIC PROGRAMMING, OPTIMIZATION METHODS, EXPRESSIVE
ENCODINGS

ЗМІСТ

Перелік умовних позначень, скорочень і термінів	7
Вступ.....	9
1 Методи та підходи вирішення задач класифікації	11
1.1 Задача класифікації: визначення, види	11
1.2 Способи вирішення задачі класифікації.....	12
1.3 (Назва третього підрозділу)	14
Висновки до розділу 1.....	15
2 (Назва другого розділу)	16
2.1 (Якийсь підрозділ).....	16
2.2 (Якийсь наступний підрозділ з дуже-дуже довгою назвою, загальна кількість слів в якій, однак, не повинна перевищувати 12 слів)	17
Висновки до розділу 2.....	18
3 (Назва третього розділу)	19
3.1 (якийсь підрозділ)	19
Висновки до розділу 3.....	20
Висновки	21

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ML — машинне навчання (англ. Machine Learning)

MLP — багатошаровий перцептрон (англ. Multilayer Perceptron)

EA — еволюційний алгоритм (англ. Evolutionary Algorithm)

GP — генетичне програмування (англ. Genetic Programming)

Adam — адаптивна оцінка моменту (англ. Adaptive Moment Estimation)

$(1 + \lambda)$ -EA with GP encodings — еволюційний алгоритм, який може використовуватися для вирішення задач класифікації (англ. $(1 + \lambda)$ -Evolutionary Algorithm with Genetic Programming encodings).

MLP with gradient descent — багатошаровий перцептрон, який використовує метод на основі градієнтного спуску, в якості оптимізаційного алгоритму.

MLP with single-point mutation — багатошаровий перцептрон, який використовує одноточкову мутацію, в якості оптимізаційного алгоритму.

ReLU — випрямлений лінійний вузол (англ. Rectified Linear Unit).

Індивід — це об'єкт, який містить в собі всю необхідну інформації для потенційного вирішення задачі.

Популяція — це набір індивідів.

Фітнес-функція — функція, яка вимірює наскільки добре пристосований кожен індивід в популяції, до зовнішньої середовища.

Кросинговер — операція в якій генерується новий індивід шляхом, якогось комбінування двох індивідів з минулого покоління (батьків).

Мутація — якась модифікація індивіду.

Контрольованість у контексті $(1 + \lambda)$ -еволюційного алгоритму з генетичним програмуванням — визначається як здатність алгоритму дозволяти користувачу точно регулювати його параметри (наприклад,

кількість нащадків λ і стратегії мутації), щоб оптимізувати процес пошуку рішення та адаптувати його під специфічні умови задачі.

ВСТУП

Актуальність дослідження. Використання класифікаційних задач має широкий спектр застосування в різних сферах наукових досліджень та практичних областях. Наприклад, важливо класифікувати, чи особа є носієм певного захворювання, спираючись на ретгенівські знімки або аналізи крові, що дозволяє з високою точністю визначати наявність патологій. З цього випливає необхідність дослідження алгоритмів, які вирішують задачу класифікації, для того, щоб мати більшу гнучкість у налаштуванні процесу пошуку по певним областям простору рішень. Відповідно до цього актуальність дослідження полягає у порівнянні алгоритмів MLP та $(1 + \lambda)$ -EA with GP encodings, щоб перевірити чи надає останній можливість контролювати гіперпараметри для більш детального пошуку та можливість краще адаптуватися до поточної задачі.

Метою дослідження є пошук оптимального методу класифікації серед методів MLP with gradient descent, MLP with single-point mutation, $(1 + \lambda)$ -EA with GP encodings, для дослідження контрольованості (див. означення у розділі перелік умовних позначень, скорочень і термінів) у розв'язанні задач бінарної та мультикласової класифікації табличних даних та картинок.

Об'єктом дослідження є якісна поведінка MLP та $(1 + \lambda)$ -еволюційних алгоритмів для задачі бінарної та мультикласової класифікації.

Предметом дослідження є особливості контролювання алгоритмів на прикладі MLP with gradient descent, MLP with single-point mutation, $(1 + \lambda)$ -EA with GP encodings на прикладі застосування до задач бінарної та мультикласової класифікації табличних даних та картинок.

Наукова новизна полягає в дослідженні та порівнянні алгоритмів MLP with gradient descent, MLP with single-point mutation, $(1 + \lambda)$ -EA with

GP encodings на прикладі задач бінарної та мультикласової класифікації.

Практичне значення результатів полягає в використанні перелічених вище методів, для задачі класифікації, для покращення контрольованості і збереженню такої ж точності та швидкості, як і в класичних методах.

1 МЕТОДИ ТА ПІДХОДИ ВИРІШЕННЯ ЗАДАЧ КЛАСИФІКАЦІЇ

В даному розділі будуть основні теоретичні відомості про об'єкт дослідження та огляд суміжних робіт в даній сфері.

1.1 Задача класифікації: визначення, види

Класифікація — це процес віднесення об'єкту до певної категорії або класу на основі його характеристик, серед заздалегідь встановленого набору категорій. Класифікація може бути бінарною, багатокласовою, багатомітковою, ієрархічною та інші. Бінарна класифікація - це класифікація, коли кожному об'єкту обирається група з наперед визначеної множини груп в якій знаходиться рівно дві групи; багатокласова класифікація - це класифікація, коли кожному об'єкту обирається група з наперед визначеної множини груп в якій може знаходитися довільна кількість груп. В поточній роботі ми зосередимося на бінарній та багатокласовій класифікації.

Задача класифікації зустрічається в багатьох сферах, наприклад: медицина (діагностика раку на основі зображень МРТ), фінанси (класифікація позичальників як „надійних“ чи „ризикованих“ на основі їхньої кредитної історії), роздрібна торгівля (класифікація покупців за типами покупок для надання персоналізованих знижок), транспорт (розрізнення між легковими авто, вантажівками та мотоциклами на дорозі), освіта (ідентифікація студентів, яким потрібна додаткова допомога в певних предметах), безпека (класифікація електронних листів як „безпечні“, „спам“ або „фішинг“), біотехнології (розпізнавання мутацій, що спричиняють хвороби).

1.2 Способи вирішення задачі класифікації

Існує декілька способів для вирішення задачі класифікації: класичні статистичні методи (наприклад, логістична регресія [4]), алгоритми машинного навчання (наприклад, метод k-найближчих сусідів [2]), глибинне навчання (за допомогою нейронних мереж), а також задачу класифікації можна вирішувати за допомогою генетичних алгоритмів. Розглянемо більш детально, як працює глибинне навчання та генетичні алгоритми, розглядати детально, як працюють алгоритми класичного навчання ми не будемо, оскільки вони дуже різні, тому важко описати загальну концепцію, як вони працюють, оскільки в кожного алгоритму вона своя, але в наступному розділі я надам перелік основних алгоритмів та посилання на статті, де про них можна прочитати.

Почнемо розгляд з глибинного навчання. В глибинному навчанні алгоритмами для вирішення задач є нейронні мережі. Нейромережі бувають дуже багатьох видів, але ми будемо їх розглядати на прикладі багат шарового перцептрон, оскільки саме його ми і використовуємо для експериментів. Багат шаровий перцептрон складається з шарів нейронів. Кожен нейрон в шарі, приймає якісь вхідні дані з попереднього шару, та обчислює вихідний сигнал, який передається наступному шару. Загалом те, що робить один нейрон можна описати наступною формулою:

$$a = f \left(\sum_{i=1}^n w_i x_i + b \right) \quad (1.1)$$

де x_1, x_2, \dots, x_n — вхідні сигнали до нейрону; w_1, w_2, \dots, w_n — ваги, що призначені для кожного вхідного сигналу; b — зсув (англ. bias), що додається до суми зважених вхідних сигналів; f — активаційна функція, що може бути сигмоїдою, ReLU, тангенсом гіперболічним та іншими. В нейронній мережі може бути довільна кількість шарів та в кожному шарі може бути довільна кількість нейронів. Але усі вони працюють по вище наведеному принципу: на вхід кожному нейрону в кожному шарі

приходить сигнал з попереднього шару і кожний нейрон генерує вихід, якщо це перший шар, то на вхід даються самі дані. Загалом схема нейронної мережі може виглядати наступним чином рисунок 1.1 в якому кожний круг це один нейрон, який розраховує функцію (1.1).

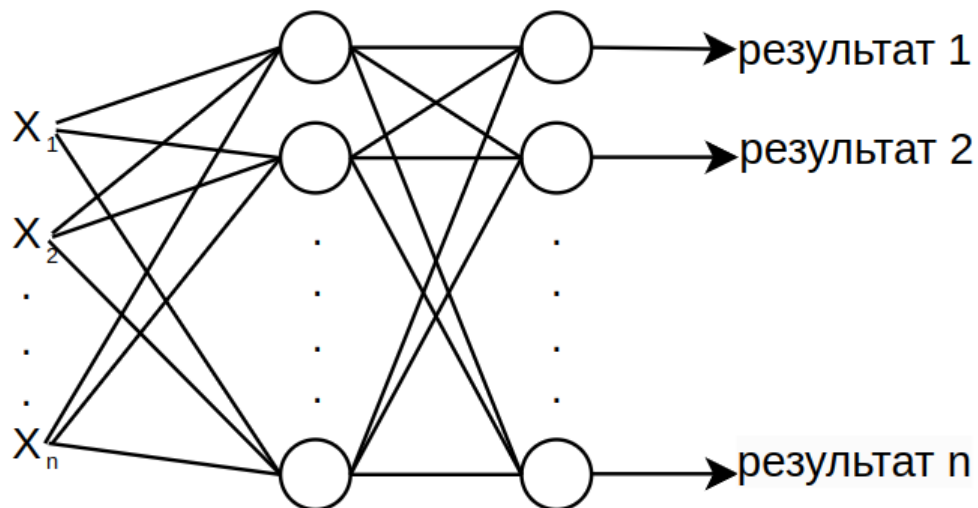


Рисунок 1.1 – Загальна архітектура повнозв'язної нейронної мережі

Тепер розглянемо, як працюють генетичні алгоритми. Існує багато різних варіацій генетичних алгоритмів, але основна структура наступна: першим кроком ініціалізується популяція індивідів, для задачі класифікації, кожен індивід буде класифікатором, далі розраховується фітнес функція для кожного індивіду, далі якимось чином обирається підмножина індивідів для створення наступного покоління, далі застосовуються операції мутації та кросинговеру до відібраних індивідів, далі розраховується фітнес функція для кожного нового індивіду, далі відбираються індивіди для нової популяції. Цей процес повторюється ітеративно, певну кількість ітерацій, або поки не буде виконана якась умова завершення алгоритму. Індивід може бути представлений різними способами, наприклад вектором бітів. Фітнес функція може варіюватися від задачі, наприклад, якщо індивід це вектор бітів, а задача це отримати вектор тільки з одиниць, то в якості фітнес функції може бути функція, яка розраховує кількість одиниць в векторі (в цьому випадку, чим більша

фітнес функція тим більше пристосований індивід). Методів відбору для репродукції існує також багато, але ось декілька прикладів найпопулярніших: рулетковий відбір [3] (кожному індивіду присвоюється ймовірність з яким він може обраться для репродукції і після цього обираються індивіду, враховуючі ці ймовірності), турнірний відбір [1] (випадковим чином обирається група індивідів з усієї популяції і з цієї групи для репродукції обирається той індивід у якого найкраща фітнес функція), елітизм

1.3 (Назва третього підрозділу)

Надамо деякі рекомендації щодо використання даного стилізового файлу.

Теорема 1.1. Використовуйте оточення *theorem* для теорем.

Доведення. Для доведень використовуйте оточення *proof*. □

Теорема 1.2. Нумерація відбувається автоматично

Твердження 1.1. Використовуйте оточення *claim* для тверджень.

Лема 1.1. Використовуйте оточення *lemma* для лем.

Наслідок 1.1. Використовуйте оточення *corollary* для наслідків.

Означення 1.1. Використовуйте оточення *definition* для визначень.

Приклад 1.1. Використовуйте оточення *example* для прикладів, на які є посилання.

Зауваження. Використовуйте оточення *remark* для зауважень. Зверніть увагу, як веде себе команда **emph**

Висновки до розділу 1

Наприкінці кожного розділу ви повинні навести коротенькі підсумки по його результатах. Зокрема, для оглядового розділу в якості висновків необхідно зазначити, які задачі у даній тематиці вже були розв'язані, а саме поставлена вами задача розв'язана не була (або розв'язана погано), тому у наступних розділах ви її й розв'язуєте.

Якщо ваш звіт складається з одного розділу, пропускайте висновок до нього – він повністю включається в загальні висновки до роботи

2 (НАЗВА ДРУГОГО РОЗДІЛУ)

До другого розділу також краще написати малесенький вступ. Зокрема, це збільшує загальний об'єм роботи та покращує її читабельність.

2.1 (Якийсь підрозділ)

У другому розділі необхідно наводити розв'язання поставленої перед вами задачі у теоретичному або аналітичному сенсі (хоча, звісно, все залежить від того, яка саме задача перед вами поставлена).

Для подання матеріалів можна використовувати таблиці (наприклад, Таблицю 2.1). Розмір шрифту у таблиці може бути меншим за 14 pt (наприклад, 12 pt, або навіть 10 pt, якщо так таблиця виглядає зрозуміліше та компактніше).

Таблиця 2.1 – Розрахунок якоїсь фантастичної дичини у декілька кроків

Параметр x_i	Параметр x_j				Перший крок		Другий крок	
	X_1	X_2	X_3	X_4	w_i	K_{bi}	w_i	K_{bi}
X_1	1	1	1.5	1.5	5	0.31	19	0.32
X_2	1	1	1.5	1.5	5	0.31	19	0.32
X_3	0.5	0.5	1	0.5	2.5	0.16	9.25	0.16
X_4	0.5	0.5	1.5	1	3.5	0.22	12.25	0.20
Разом:					16	1	59.5	1

Бажано, щоб кожен пункт завдань, окреслених у вступі, відповідав певному розділу або підрозділу у дипломній роботі.

Теорема 2.1. *Нумерація у наступних розділах також*

представляється автоматично та коректно.

2.2 (Якийсь наступний підрозділ з дуже-дуже довгою назвою, загальна кількість слів в якій, однак, не повинна перевищувати 12 слів)

Для подання матеріалів також дуже зручними є рисунки (наприклад, рисунки 2.1 або 2.2).

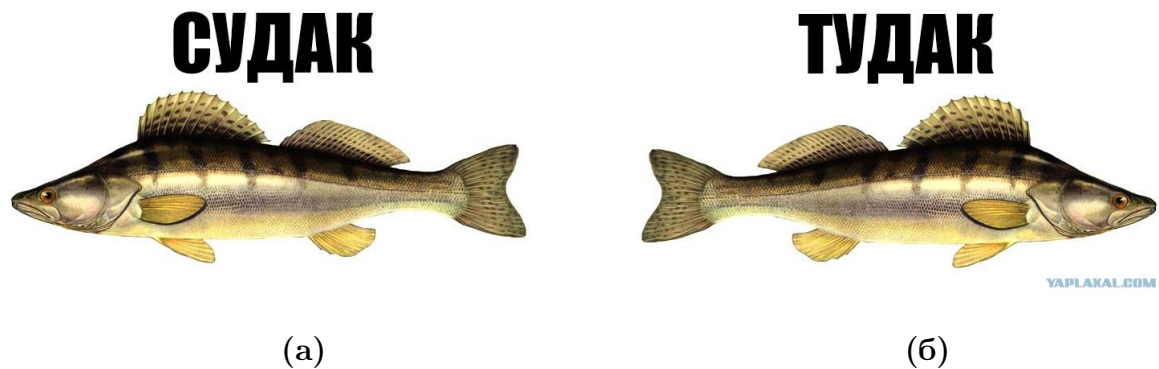


Рисунок 2.1 – Різні види риб: (а) судак, (б) тудак.

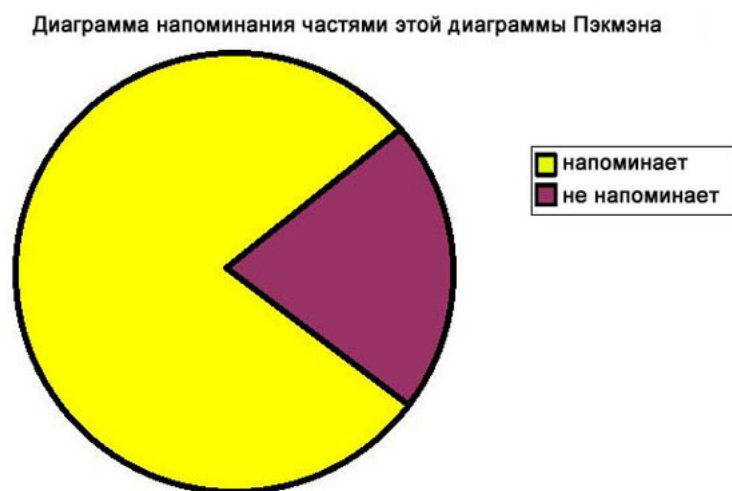


Рисунок 2.2 – Частка кругових діаграм, які схожі на Пекмена

Висновки до розділу 2

Наприкінці розділу знову наводяться коротенькі підсумки.

3 (НАЗВА ТРЕТЬОГО РОЗДІЛУ)

3.1 (якийсь підрозділ)

Подивіться, як нераціонально використовується простір, якщо не писати вступи до розділів. :)

Зазвичай третій розділ присвячено опису практичного застосування або експериментальної перевірки аналітичних результатів, одержаних у другому розділі роботи. Втім, це не обов'язкова вимога, і структура основної частини диплому більш суттєво залежить від характеру поставлених завдань. Навіть якщо у вас є певне експериментальне дослідження, але його загальний опис займає дві сторінки, то краще приєднайте його підрозділом у попередній розділ.

При описі експериментальних досліджень необхідно:

- наводити повний опис експериментів, які проводились, параметрів обчислювальних середовищ, засобів програмування тощо;
- наводити повний перелік одержаних результатів у чисельному вигляді для їх можливої перевірки іншими особами;
- представляти одержані результати у вигляді таблиць та графіків, зрозумілих людському оку;
- інтерпретувати одержані результати з точки зору поставленої задачі та загальної проблематики ваших досліджень.

У жодному разі не потрібно вставляти у даний розділ тексти інструментальних програм та засобів (окрім того рідкісного випадку, коли саме тексти програм і є результатом проведення експериментів). За необхідності тексти програм наводяться у додатках.

Висновки до розділу 3

Висновки до останнього розділу є, фактично, підсумковими під усім дослідженням; однак вони повинні стосуватись саме того, що розглядалось у розділі.

ВИСНОВКИ

Загальні висновки до роботи повинні підсумовувати усі ваші досягнення у даному напрямку досліджень.

За кожним пунктом завдань, поставлених у вступі, у висновках повинен міститись звіт про виконання: виконано, не виконано, виконано частково (І чому саме так). Наприклад, якщо першим поставленим завданням у вас іде «огляд літератури за тематикою досліджень», то на початку висновків ви повинні зазначити, що «у ході даної роботи був проведений аналіз опублікованих джерел за тематикою (...), який показав, що (...)». Окрім простої констатації про виконання ви повинні навести, які саме результати ви одержали та проінтерпретувати їх з точки зору поставленої задачі, мети та загальної проблематики.

В ідеалі загальні висновки повинні збиратись з висновків до кожного розділу, але ідеал недосяжний. :) Однак висновки не повинні містити формул, таблиць та рисунків. Дозволяється (та навіть вітається) використовувати числа (на кшталт «розроблена методика дозволяє підвищити ефективність пустопорожньої балаканини на 2.71%»).

Наприкінці висновків необхідно зазначити напрямки подальших досліджень: куди саме, як вам вважається, необхідно прямувати наступним дослідникам у даній тематиці.

ЛІТЕРАТУРА

- [1] Yongsheng Fang та Jun li. “A Review of Tournament Selection in Genetic Programming”. В: жовт. 2010, с. 181—192. ISBN: 978-3-642-16492-7. DOI: 10.1007/978-3-642-16493-4_19.
- [2] Gongde Guo та ін. “KNN Model-Based Approach in Classification”. В: *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*. За ред. Robert Meersman, Zahir Tari та Douglas C. Schmidt. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, с. 986—996. ISBN: 978-3-540-39964-3.
- [3] Adam Lipowski та Dorota Lipowska. “Roulette-wheel selection via stochastic acceptance”. В: *Physica A: Statistical Mechanics and its Applications* 391 (вер. 2011). DOI: 10.1016/j.physa.2011.12.004.
- [4] Joanne Peng, Kuk Lee та Gary Ingersoll. “An Introduction to Logistic Regression Analysis and Reporting”. В: *Journal of Educational Research - J EDUC RES* 96 (вер. 2002), с. 3—14. DOI: 10.1080/00220670209598786.