МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

П.П. Маслянко

Системи Data Science

Методичні вказівки та рекомендації для виконання лабораторних робіт

Спеціальність 113 Прикладна математика
Спеціалізація - Наука про дані та математичне моделювання

Ухвалено кафедрою прикладної математики
(протокол № _13_ від _16.06.22_)
Погоджено Методичною комісією факультету прикладної математики[1]
(протокол № _9_ від _24.06.22_)

Київ – 2022

[1]Методичною радою університету– для загальноуніверситетських дисциплін.

ЗМІСТ

Системи DS

Короткі методичні вказівки та рекомендації для виконання лабораторних робіт № 1,2,3.

1. **Основне завдання - Інженерія системи DS для вирішення завдань діяльності Організаційної системи (типової задачі DS).**

Основні категорії інженерії систем DS:

- бізнес-модель, бізнес-процеси;
- завдання бізнес-моделювання;
- категорії теорії систем;
- категорії математичної статистики;
- категорії DS, ML,DL;
- інші інструменти аналізу даних та моделювання.

2. Типові задачі Data Science для вирішення завдань діяльності Організаційної системи

[Provost Foster, and Tom Fawcett. Data Science for Business: What you need to know about data mining and data-analytic thinking. " O'Reilly Media, Inc.", 2013. Mode of access.]

1. Задачі навчання з вчителем (**Supervised learning**):

1.1. Задачі класифікації множини сутностей і оцінювання вірогідності належності сутності до певного класу. Ієрархічний і фасетний методи класифікації. Побудова моделі класифікатора і класифікатора для визначеної множини сутностей.

Візуалізація і документування результатів.

Висновки за результатами вирішення задачі належності сутності до певного класу.

1.2. Задачі регресії, тобто прогнозування значення залежної змінної за допомогою незалежної і визначення внеску окремих незалежних змінних у варіацію залежної змінної.
Візуалізація і документування результатів.
Висновки за результатами вирішення задачі прогнозування значення залежної змінної.

2. Задачі навчання без вчителя (**Unsupervised learning**):

2.1. Задачі кластеризації, тобто розбиття заданої множини сутностей на підмножини-кластери за визначеними ознаками так, щоб кожен кластер складався з подібних сутностей за встановленою ознакою, а сутності різних кластерів суттєво відрізнялися один від одного.
Візуалізація і документування результатів.
Висновки за результатами вирішення задачі розбиття заданої множини сутностей на підмножини-кластери за визначеними ознаками.

2.2. Задачі визначення подібних (або діаметрально протилежних) сутностей (явних або прихованих) на основі даних про ці сутності. Групування подібних (або діаметрально протилежних) сутностей у потрібні конфігурації (ієрархії, області).
Візуалізація і документування результатів.
Висновки за результатами вирішення задачі визначення подібних (або діаметрально протилежних) сутностей (явних або прихованих) на основі даних про ці сутності.

2.3. Задачі групування за збігами, тобто пошук подібних сутностей, що проявляються при проведенні певної діяльності з цими сутностями. Проведення певних операцій (транзакцій) з сутностями визначеного класу спонукає появу поряд з ними визначених сутностей іншого класу.
Візуалізація і документування результатів.

Висновки за результатами вирішення задачі виявлення збігу появи одних сутностей поряд з іншими при проведенні певної діяльності з останніми.

2.4. Задачі прогнозування і ітеративний процес виявлення відношень між сутностями на основі даних про ці сутності, тобто виявлення явних, або прихованих відношень (зв'язків), статичний і динамічний характер цих відношень, а також кількісну та якісну міру відношень.
Візуалізація і документування результатів.
Висновки за результатами вирішення задачі прогнозування і ітеративного процесу виявлення відношень між сутностями на основі даних про ці сутності.

2.5. Задачі профілювання (моделювання типової поведінки індивіда, групи індивідів, популяції). Профілювання вимагає моделювання поведінки з різних точок зору з наступною композицією/інтеграцією всіх отриманих результатів. Надзвичайно актуальна задача для вивчення закономірностей поведінки людської спільноти в умовах соціально-економічної невизначеності і ризиків.
Візуалізація і документування результатів.
Висновки за результатами вирішення задачі профілювання (моделювання типової поведінки індивіда, групи індивідів, популяції).

3. Задачі семплювання, тобто цілеспрямована заміна (проріджування) великої кількості даних на меншу кількість даних для отримання кращих результатів і зменшення ресурсів на вирішення задачі.
Візуалізація і документування результатів.
Висновки за результатами вирішення задачі семплювання.

4. Задачі причинно-наслідкового моделювання, тобто прогнозування того, які дії (чи бездіяльність) спричиняють інші дії чи стан. При цьому

надзвичайно важливо дуже чітко вказати припущення при яких причини і відповідні наслідки справджуються.

Візуалізація і документування результатів.

Висновки за результатами вирішення задачі причинно-наслідкового моделювання і припущень при яких причини і відповідні наслідки справджуються.

3. Де знайти відкриті дані ???

1. Портал відкритих даних UA https://data.gov.ua/
2. Відкриті дані – Портал Києва – КМДА *kyivcity.gov.ua › vidkriti_dani_257959*
3. Чорний ринок даних з посиланнями. https://www.epravda.com.ua/columns/2020/12/30/669656/#comments
4. Державна служба статистики http://www.ukrstat.gov.ua/
5. https://www.kaggle.com/
6. Habr – Корисний Смітник https://habr.com/ru/company/mailru/blog/462769/
7. Інші ресурси за вибором студента **з посиланням**!!!

4. Процеси Data Science

1. **Приведемо базовий приклад процесу Data Science**, запропонований Кеті О'Ніл та Рейчел Шатт [3]:

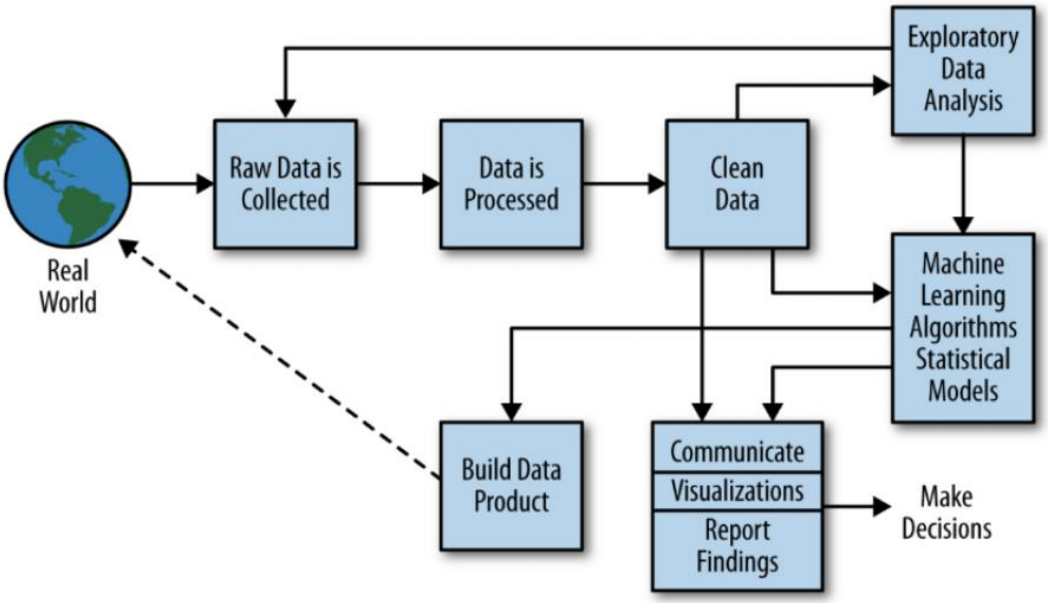


Рисунок 4 — Процес Data Science

[O'Neil, Cathy, and Rachel Schutt. Doing data science: Straight talk from the frontline. " O'Reilly Media, Inc.", 2013.]

В даному представленні виділені 8 основних етапів процесу:

1) Збір сирих даних з будь-яких ресурсів реального світу.
2) Обробка даних.
3) Їхня очистка.
4) Розвідувальний аналіз даних.
5) Побудова статистичних моделей та застосування алгоритмів Machine Learning із використанням зібраних, оброблених, очищених даних.
6) Етап внутрішньої комунікації. Представлення результатів членам команди розробників конкретної системи Data Science, а також її зацікавленим сторонам.
7) Створення / продукування нових даних, що застосовуються в реальному світі.
8) Прийняття рішень на основі отриманих результатів.

Варто відзначити наявність прямих зв’язків між певними етапами: наприклад, розвідувальний аналіз може виявити нестачу даних, які необхідно

додатково зібрати, або ж очищені дані можуть бути візуалізовані з метою роз'яснення їхньої природи колегам по команді.

2. **Міжгалузевий стандарт** процесу майнингу даних CRISP-DM, Shearer, 2000

Порівняємо вищеописаний процес Data Science з проаналізованим Фостером Провостом та Томом Фосеттом міжіндуструальним стандартним процесом для Data Mining (*CRISP-DM — CR*oss *I*ndustry *S*tandard *P*rocess for *D*ata *M*ining) [7], приведеним на рисунку 5.

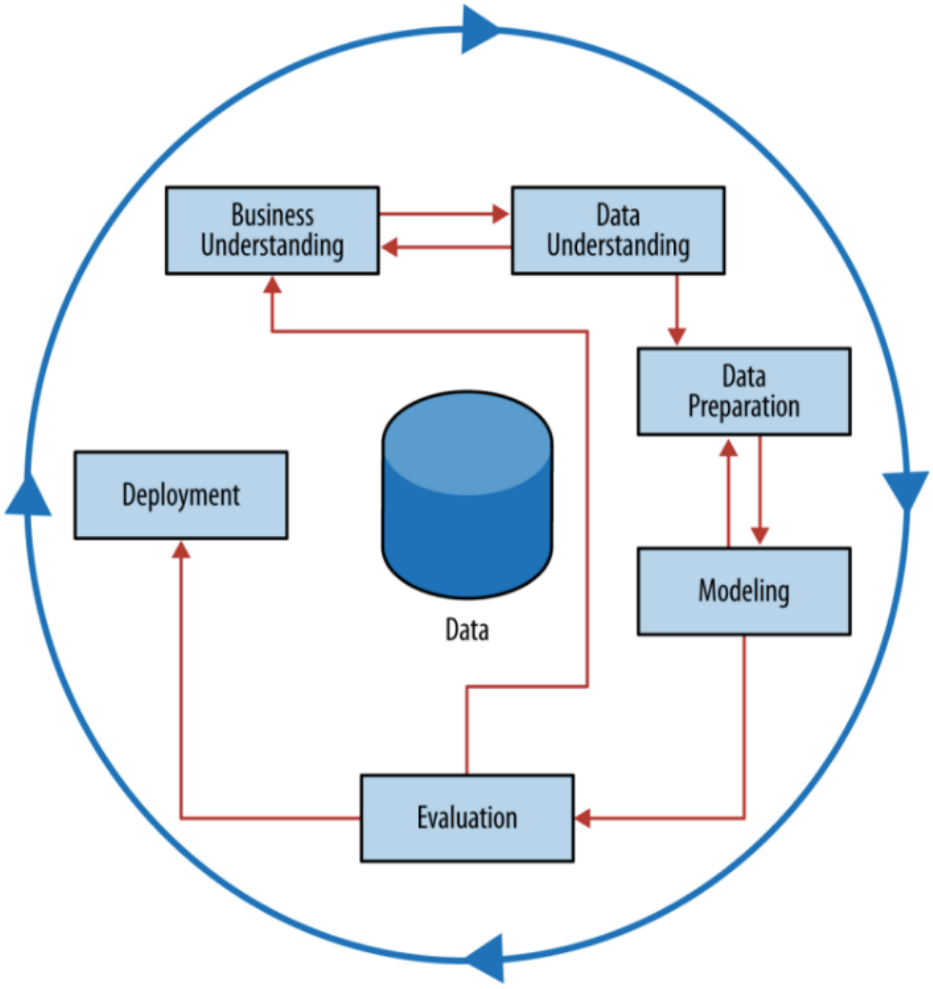


Рисунок 5 — Стандартний процес для Data Mining
[**Міжгалузевий стандарт** процесу майнингу даних CRISP-DM, Shearer, 2000]

Відзначимо одразу його кругову ітеративну природу, а також наявність двох досить абстрактних, узагальнюючих етапів — розуміння бізнесу та розуміння даних:

- Перший уособлює необхідність чіткої постановки задачі згідно з заданою проблемою, пошук креативних методів досягнення мети та оптимальної формалізації проблеми, що дозволила б застосовувати вже існуючі методи якомога ефективніше.
- Тоді як **другий етап зосереджений на стратегічному аналізі** основної сировини Data Mining — даних. Тут вкрай необхідним є розуміння основної структури, недоліків, переваг використовуваних даних, немало важливою є й оцінка потенційних джерел пошуку додаткової інформації, необхідних інвестицій (як часових, так і фінансових) в їхнє дослідження та використання та потенційної користі.

Далі — невід'ємна **процедура підготовки даних**, в якій, проводячи аналогію з процесом на рисунку 4, об'єднуються обробка, очистка та розвідувальний аналіз даних, що самі по собі можуть становити багато-ітеративний підпроцес.

Наступний етап CRISP-DM — **моделювання** — також має пряму відповідність в представленні процесу Data Science, де, знову ж таки, усі назви є більш конкретними.

Будь-які розробки повинні підпорядковуватися контролю якості шляхом проведення регулярних **верифікації і валідації** побудованих моделей та системи загалом. Автори також неявним чином наголошують на необхідній комунікації, представленні результатів, а також на особливостях їхнього простого та **зрозумілого тлумачення** ключовим зацікавленим сторонам (інвесторам), на долю яких і випадає прийняття основних бізнес-рішень.

5. Верифікація і валідація

Стандарт ICO 9000:2000 визначає ці терміни таким чином:

«**Верифікація (**verification — перевірка,**)** — підтвердження на основі надання об'єктивних свідчень того, що *встановлені вимоги були виконані*».

«**Валідація** (validation — надання законної сили)— підтвердження на основі надання об'єктивних свідчень того, що встановлені вимоги, *призначені для конкретного використання і застосування, виконані*».

Таким чином можна зробити **висновок:**

верифікація — проводиться методом порівняння значень конкретних характеристик продукції, товарів і послуг з заданими в документації значеннями цих характеристик для формування висновку про відповідність продукції, товарів і послуг заданим в документації вимогам;

валідація — проводиться при необхідності і виконується методом аналізу заданих умов застосування/експлуатації і оцінки відповідності значень характеристик продукції, товарів і послуг цим умовам, з наступним висновком про можливість застосування продукції, товарів і послуг для встановлених умов застосування/експлуатації.

6. Системна інженерія систем Data Science.

Системна інженерія систем DS реалізується на основі бізнес-профіля Еріксона-Пенкера. Шаблон такої реалізації покажемо на прикладі системної інженерії систем нейронного машинного перекладу. Повний текст цього проекту можна прочитати за посиланням [Маслянко П.П., Сєльський Є.П. Метод системної інженерії систем нейтронного машинного перекладу. *KPI Science News*, 2021, № 2. с. 46 – 55]. https://doi.org/10.20535/kpisn.2021.2.236939

Вступ

Поряд із надзвичайно широким застосуванням систем машинного перекладу (СМП) на ринку існує не так багато компаній-розробників, продукти яких мають попит. Це, зокрема, безоплатні та комерційні продукти, як-от:"*GoogleTranslate*","*DeepLTranslator*","*ModernMT*","*Apertium*" тощо. Практична потреба у просуванні на ринок якісних СМП потребує від розробників більш упорядкованих і систематизованих методів їх розроблення.

Однак у літературі практично не висвітлюють хоч якісь загально прийняті чи корпоративні стандарти й методи розроблення таких СМП. Водночас для більш ефективного та продуктивного розроблення якісних СМП потрібні науково обґрунтовані методи інженерії СМП, щоб якнайшвидше отримати якісний і конкурентоспроможний продукт.

У цій роботі буде досліджено особливості розроблення систем нейронного машинного перекладу (СНМП), що поєднують найкращі властивості СМП на основі граматичних правил і СМП, реалізованих на статистичних алгоритмах.

Постановка задачі

Метою цієї роботи є застосування бізнес-профілю Еріксона–Пенкера [1] для розроблення та формалізації методу системної інженерії СНМП.

Методологія системної інженерії і бізнес-профіль Еріксона–Пенкера

Основна ідея методу системної інженерії СНМП полягає у застосуванні методології системної інженерії та бізнес-профіля Еріксона–Пенкера для формалізації упорядкованого способу розроблення СНМП.

Методологія системної інженерії базується на трьох основних категоріях [2]:

1. Категорія “Система” як множина сутностей і відношень між ними, що в межах прийнятих припущень й обмежень показує, власне, систему.

2. Категорія “Життєвий цикл”, що передбачає представлення генезису системи від народження та до утилізації самої системи [2].

3. Категорія “Зацікавлені сторони”, що передбачає формування вичерпної множини умов і вимог, які зацікавлені сторони висувають до системи.

Проаналізуємо можливість застосування технік системної інженерії, яку також називають системною методологією XXI століття (за Дереком Хітчінсом [3]). Однією з найпоширеніших моделей представлення діяльності є бізнес-

профіль Еріксона–Пенкера [1], в контексті якого автори сформулювали чотири основні сутності формального представлення діяльності будь-якої бізнес-системи:

– *цілі* (уособлюють мету діяльності системи та сформульовані як правило. Цілі можуть бути розбиті на підцілі та досягненізавдяки реалізації процесів);

– *процеси* (основні дії, що складають діяльність системи та призначені для досягнення мети відповідно до встановлених бізнес-правил. Процеси зазвичай підпорядковуються правилам, можуть змінювати стан вхідних ресурсів, а також продукувати нові ресурси – ресурси виходу системи згідно з умовами та вимогами, встановленими зацікавленими особами);

– *ресурси* (фізичні, абстрактні чи інформаційні об’єкти, які система споживає, використовує, обробляє та продукує впродовж всієї своєї діяльності для досягнення мети);

– *правила* (певні формалізовані обмеження, рамки, умови та вимоги тощо, що накладаються на процеси, а також описують характер зв’язків між ресурсами).

Основні діаграми, необхідні для формального графічного представлення та моделювання систем на основі бізнес-профіля Еріксона–Пенкера, поділяють на два основні типи:

– діаграми структурного представлення, а саме:

1. *Діаграму класів* (ієрархічний/логічний показ зв’язків ізалежностей, наявних між класами сутностей, що складають систему. Під “класом сутностей” системи маємо на увазі ту чи іншу множину однієї з чотирьох сутностей, описаних вище);

2. *Діаграму компонентів* (схема поділу системи на частини –*компоненти*, що їх формують за певною, найчастіше функціональною, ознакою. Компоненти можуть реалізовувати один чи більше процесів і взаємодіяти з одним чи більше ресурсами. На діаграмі компонентів також вказують відношення між

компонентами, реалізованими як формалізовані правила-*інтерфейси* для забезпечення взаємодії компонентів у системі);

– діаграми динамічного представлення, а саме:

1. *Діаграму діяльності*(зображує поетапний перебіг процесів системи, що пов'язані один з одним через вхідні та вихідні ресурсів; на діаграмі діяльності обов'язково зазначають її початок і кінець);

2. *Діаграму процесів із "водними доріжками"*(поєднує представлення процесів діяльності з компонентами системи, що реалізують тучи іншу діяльність. Тобто будь-який процес має перебувати в межах свого батьківського компонента (відповідної вертикальної смуги, водної доріжки). Міжкомпонентні інтерфейси також можуть бути включені до цієї діаграми).

Така впорядкована множина формалізованих (зокрема, в нотації UML) сутностей і представлень системи на основі бізнес-профіля Еріксона–Пенкера є повною моделлю діяльності бізнес-системи, що враховує вимоги всіх зацікавлених осіб.

Метод системної інженерії СНМП

Під терміном "метод" тут і надалі у статті ми розуміємо "систематизований спосіб досягнення теоретичного чи практичного результату, розв'язання проблем чи одержання нової інформації на основі певних регулятивних принципів та дії, усвідомлення специфіки досліджуваної предметної галузі та законів функціонування її об'єктів" [4].

Метод розроблення СНМП (надалі – метод) на основі застосування технік системної інженерії складається з трьох основних етапів.

Формалізація структурного представлення СНМП. На першому етапі структуру СНМП моделюють як бізнес-профіль Еріксона–Пенкера [1, 5] (рис. 1): визначають її проблему, мету, ресурси, процеси, цілі, правила; проектують структурне представлення в контексті, означеннях і моделях представлень області СМП з урахуванням інтересів усіх зацікавлених сторін.

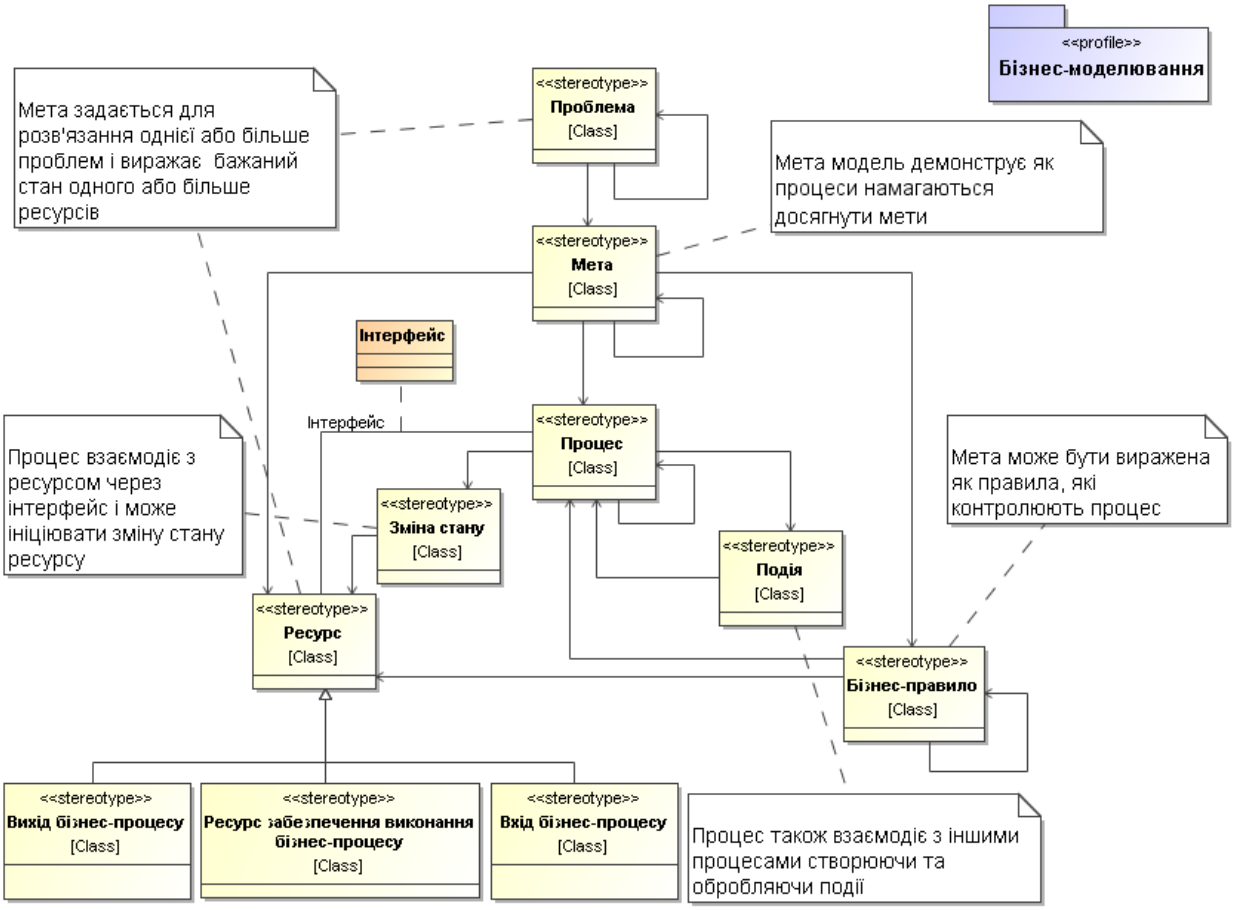


Рис. 1.Удосконалений бізнес-профіль Еріксона–Пенкера. Діаграма класів у нотації UML [5]

Визначимо зміст кожного з класів діаграми (див. рис. 1) у термінах постановки задачі інженерії СНМП, а саме класах:

1. *Проблема* (актуальне питання, що потребує відповідних рішень, основна мотивація розроблення СНМП, яка спонукає до формулювання конкретної мети. Проблема цієї роботи: *необхідність якісного англійсько-українського перекладу)*;

2. *Мета* (виражає глобальну ціль роботи, покликану розв’язати поставлену проблему. Мета цієї роботи: *розроблення конкурентоспроможного англійсько-українського перекладача)*;

3. *Процес* (множина процесів діяльності системи, внаслідок якої досягають мети, чітко визначена послідовність дій/підпроцесів, що призводить до виконання певного завдання. Процесами цієї системи є: *Завантаження текстових даних*, *Первинне оброблення текстових даних*, *Навчання методом*

машинного перекладу (ММП), *Машинний переклад (МП)*, *Функціонування веб застосунка)*;

4. *Зміна стану* (можливі зміни певних ресурсів унаслідок роботи процесів. СНМП налічує три зміни станів: *Необроблені текстові дані→Оброблені текстові дані* (процес *Первинного оброблення текстових даних*), *Оброблені текстові дані→Перекладені текстові дані* (процес *МП*), *Ініціалізована ММП→Навчена ММП* (процес *Навчання ММП*));

Ресурс (будь-які сутності (матеріальні чи нематеріальні), що їх споживає та продукує розроблювана система. Детальнішу ієрархію ресурсів цієї системи наведено на рис. 2). Рис. 2 репрезентує компонентну модель системи англійсько-українського нейронного машинного перекладу (НМП), що відображає структуру та відношення між компонентами через реалізацію відповідних інтерфейсів.

5. Ресурси найнижчого рівня ієрархії, що беруть безпосередню участь у процесах, також поділяють за характером впливу на перебіг процесів на такі три класи:

Вихід бізнес-процесу (ресурси, що їх продукує СНМП, кінцевий результат її функціонування. До них належать *Перекладені текстові дані*, *Навчена ММП)*;

– *Ресурс забезпечення виконання бізнес-процесу* (ресурси, що забезпечують виконання процесів, але не є кінцевим результатом роботи: *Оброблені текстові дані*, *Веб застосунок*, *Модель текстового оброблення)*;

– *Вхід бізнес-процесу* (первинні ресурси входу початкових процесів, які ініціалізують цикл роботи системи: *Необроблені текстові дані*, *Ініціалізована ММП)*;

6. *Подія*(виникає через певні зовнішні фактори чи як результат взаємодії між процесами. Потенційними подіями цієї системи вважають зміну навчальних текстових даних, що впливає на *Навчання ММП*; появу нової найефективнішої версії *Навченої ММП* у процесі *Навчання ММП*, яка, як наслідок, замінить поточну версію, що бере участь у *МП*; введення текстових

даних користувачем (запит користувача) в процесі *Функціонування веб застосунка*, що спонукає систему до їх негайного перекладу);

7. *Бізнес-правило* ((з англ.–*Business Rule (BR)* формальні інструкції, що регулюють, обмежують, встановлюють контекст і рамки функціонування процесів. Приклад бізнес-правила СНМП: розмір введеного користувачем тексту для перекладу у веб застосунку не може перевищувати 500 слів).

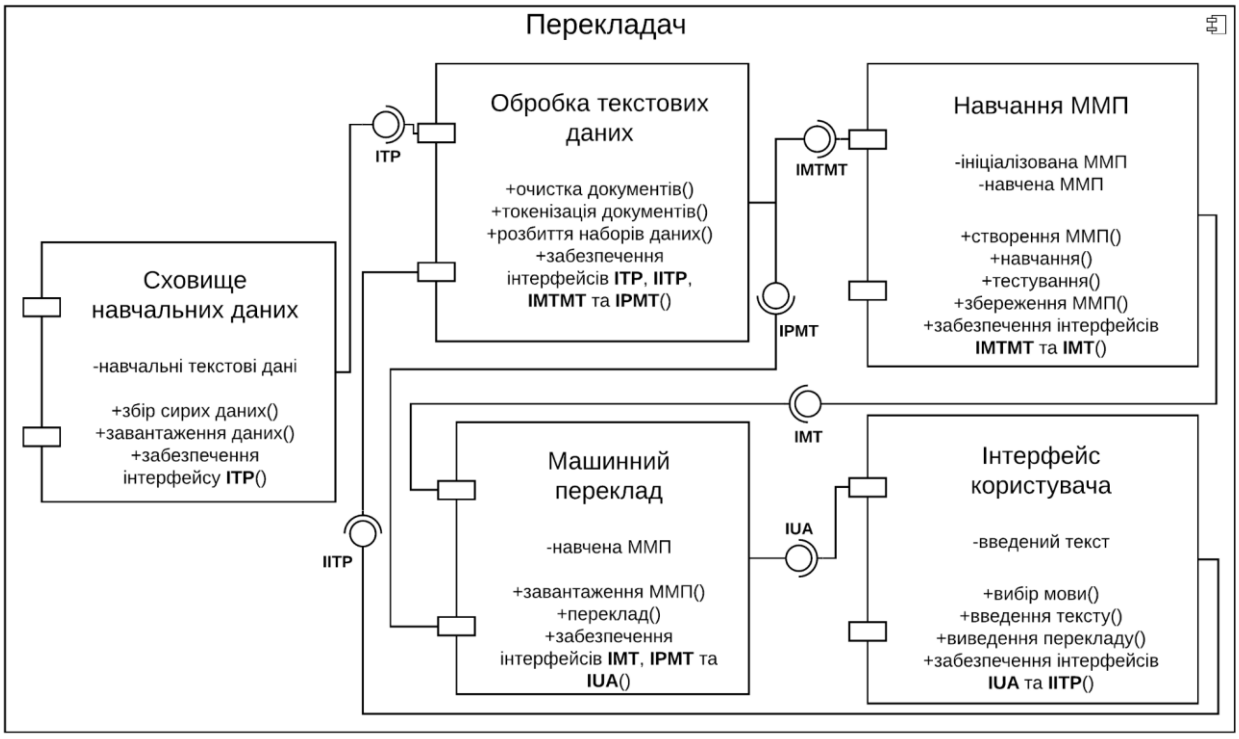


Рис. 2. Модель СНМП. Діаграма компонентів у нотації UML

Формалізація динамічного представлення СНМП. На другому етапі визначають множину процесів, характерну саме для класу систем DataScience, згідно з визначеними О'Ніл і Шатт процесом Data Science [6] і міжнародним стандартом CRISP-DM в інтерпретації Фостера та Фосетта [7]. На цьому рівні організація діяльності може бути уточнена з урахуванням особливостей СНМП та, як наслідок, декомпонована на такі три підетапи:

1.1. Збір, аналіз й оброблення навчальних текстових даних, що їх використовуватимуть під час тренування нейронних мереж НМП відповідно до моделі процесу Data Science, запропонованої О'Ніл і Шатт [6], або стадії *Data understanding* і *Data processing* міжіндуструального стандарту процесів Data

Mining (*CRISP-DM — CR*oss *I*ndustry *S*tandard *P*rocess for *D*ata *M*ining) проаналізованого Фостером та Фосеттом [7].

1.2. Власне побудова (розроблення архітектури) та навчання нейронних мереж НМП– це аналог етапу *Machine Learning Algorithms Statistical Models* [6] чи стадії *Modeling* стандартного процесу для DataMining [7].

1.3. Визначення метрик оцінювання ефективності як навчених моделей НМП, так і роботи системи загалом – аналог етапу *Report Findings* [6] чи стадії *Evaluation* стандартного процесу для Data Mining [7].

Таким чином бізнес-профіль Еріксона–Пенкера є системою класів і відношень між ними, необхідних і достатніх для представлення та розроблення СНМП. Вичерпний перелік бізнес-правил (технічних умов) до бізнес-профілю регламентує функціональність інтерфейсу користувача конкретної СНМП, наприклад:

BR1 –Розмір введеного користувачем тексту для перекладу в Інтерфейсі користувача не може перевищувати 500 слів;

BR2 – Швидкість перекладу одного запиту користувача не повинна перевищувати 5 секунд;

BR3 – Графічний інтерфейс системи налічує 2 основні текстових поля: активне поле введення тексту мовою оригіналу, з яким взаємодіє користувач, а також поле виведення перекладеного тексту цільовою мовою;

BR4 – Функція зміни мов має бути імплементована в єдиній кнопці графічного інтерфейсу, враховуючи бінарний характер вибору.

Відповідна множина технологій Data Science є інструментами імплементації класів бізнес-профіля Еріксона–Пенкера.

Верифікація та валідація методу системної інженерії СНМП. На третьому етапі проводять верифікацію та валідацію розробленої СНМП задля перевірки дотримання всіх технічних умов і вимог зацікавлених сторін. Потрібно насамперед враховувати модель МП, яку застосовують для реалізації компонента ММП. Нею може бути один із типів архітектур нейронних мереж (цей перелік не є вичерпним):

– рекурентні нейронні мережі типу кодувальник/декодувальник із механізмом уваги [8];

– згорткові нейронні мережі з наскрізними зв’язками та згорткові нейронні мережі уваги [9];

–повнозв’язні нейронні мережі типу трансформер [10].

Далі можна провести аналіз функціональності, адекватності та продуктивності розробленої СМП на основі вибраної метрики порівняльного аналізу.

Тож, спираючись на означення поняття “метод” [4], можна сформулювати означення Методу системної інженерії СНМП: **це множина класів завдань, процесів, ресурсів, бізнес-правил і відношень між ними необхідних для продукування СНМП на основі методології системної інженерії, бізнес-профіля Еріксона–Пенкера та технологій Data Science.**

Імплементація методу системної інженерії СНМП

Далі покажемо застосування методу для реалізації СНМП.

Імплементація структурного представлення СНМП. Модель СНМП відповідно до представлень класів бізнес-профіля Еріксона–Пенкера (див. рис. 2) є системою класів і відношень між ними, необхідних і достатніх для розроблення СНМП.

Функціональність і призначення компонентів СНМП:

1. *Сховище навчальних даних.* Це локальне сховище, на якому розміщено зібрані корпуси текстів, використовувані для навчання моделі машинного навчання.

2. *Оброблення текстових даних.* Будь-які первинні текстові дані насамперед уніфікують. Набір текстових даних – так званий *корпус*, що складається з текстових *документів* – текстові рядки довільної довжини. Кожен документ первинного вхідного корпусу компонента оброблення текстових даних проходить такі етапи уніфікації:

2.1. *Очищення документа* від спеціальних символів, декапіталізація слів, аналіз абревіатур. Також можливе загальне очищення корпусу від невалідних документів.

2.2. *Токенізація документа*– розбиття текстового рядка на список атомарних текстових складових –*токенів*, що в цій імплементації є окремими словами.

2.3. *Розбиття навчальних текстових даних* на відповідні набори даних (тренувальний, валідаційний, тестовий).

Інтерфейс **ІМТМТ**– фактичне передання оброблених результативних навчальних текстових даних на вхід функції розбиття навчального текстового корпусу на тренувальний, валідаційний і тестовий набори даних компонента *Навчання ММП*.

Інтерфейс **ІРМТ**– фактичне передання оброблених результативних текстових даних, введених користувачем на вхід функції МП навченої моделі МП компонента МП.

3. *Навчання ММП.* Основний складник цього компонента – власне модель МП, якою може бути один із типів архітектур нейронних мереж (цей перелік не є вичерпним):

3.1. Рекурентні нейронні мережі типу кодувальник/декодувальникіз механізмом уваги [8].

3.2. Згорткові нейронні мережі з наскрізними зв’язками та згорткові нейронні мережі уваги [9].

3.3. Повнозв’язні нейронні мережі типу трансформер [10].

Детальна архітектура нейронної мережі, а також усі відповідні гіперпараметри встановлюють емпірично під час безпосереднього етапу навчання з використанням валідаційного набору текстових даних.

У компоненті *Навчання ММП* передбачено функціонал:

– створення/ініціалізації ММП вищеописаної архітектури;

– її (пере)навчання з потенційною зміною архітектури та значень навчальних гіперпараметрів;

– фінальні якісне та кількісне тестування ефективності перекладу (проводять тільки після повного циклу валідації);

– збереження навченої ММП із застосуванням алгоритму контролю версій для подальшої підтримки та вдосконалення ММП.

Тут і надалі під версіями ММП необхідно розуміти ММП, навчені на різних версіях навчальних текстових даних, що з часом можуть бути змінені задля інтерпретації нововведених синтаксичних, семантичних, орфографічних правил тієї чи іншої мови, або ж доповнені з появою нових джерел даних перекладу. Такий контроль версій сприяє регулярному оновленню ММП згідно з останніми мовними стандартами.

4. *Машинний переклад.* Компонент безпосереднього НМП оброблених текстових даних, уведених користувачем (отримані через інтерфейс **IPMT**) через їх подачу на вхід завантаженої ММП (отриманої через інтерфейс **IMT**), що запущена в режимі передбачення (без зворотного проходження й оновлення ваг ММП). Завантаження останньої найефективнішої версії ММП входить до складу цього компонента як фактична імплементація інтерфейсу **IMT**. Він передає завантажену ММП на вхід процесу МП задля проведення операції НМП оброблених текстових даних, уведених користувачем.

Інтерфейс **IUA**– фактичне передання перекладених результативних текстових даних, уведених користувачем на вхід функції виведення перекладу компонента Інтерфейс користувача.

5. Інтерфейс користувача. Компонент, який уособлює елементарний графічний інтерфейс користувача, що йому надано три основні функції:

– обрати мову введення/перекладу;

– увести текст для перекладу: введені текстові дані передають інтерфейсом **IITP** на вхід функції очищення тексту компонента Оброблення текстових даних;

– переглянути результат перекладу (отриманий через інтерфейс **IUA**) – функція *виведення перекладу()* компонента Інтерфейс користувача.

Наведемо список усіх інтерфейсів СНМП:

– **ITP** (***I**nterface **T**ext **P**rocessing*) – інтерфейс передання завантажених в оперативну пам’ять навчальних текстових даних на вхід процесу текстового оброблення;

– **IMTMT** (***I**nterface **M**achine **T**ranslation **M**odel **T**raining*) – інтерфейс передавання вихідних (із модуля текстового оброблення) опрацьованих (очищених, нормалізованих, токенізованих, нумеризованих) навчальних текстових документів на вхід процесу навчання ММП;

– **IPMT** (***I**nterface **P**rocessing-**M**achine **T**ranslation*) – інтерфейс передавання вихідних (із модуля текстового оброблення) опрацьованих (очищених, нормалізованих, токенізованих, нумеризованих) текстових даних, введених користувачем для перекладу, на вхід модуля безпосереднього НМП;

– **IMT** (***I**nterface **M**achine **T**ranslation*) – інтерфейс завантаження останньої найефективнішої версії збереженої навченої ММП для її подальшого використання в процесі НМП попередньо обробленого тексту, введеного користувачем;

– **IUA** (***I**nterface**U**ser-**A**pplication*) – інтерфейс передання результативного перекладу тексту користувача до модуля Інтерфейс користувача для його остаточного виведення;

– **IITP** (***I**nterface **I**nput **T**ext **P**rocessing*) – інтерфейс передавання тексту, введеного користувачем, до модуля первинного текстового оброблення.

Імплементація динамічного представлення системи. Діаграма діяльності (рис. 3) – це візуалізація основних циклів функціонування процесів і їхньої взаємодії першого рівня.

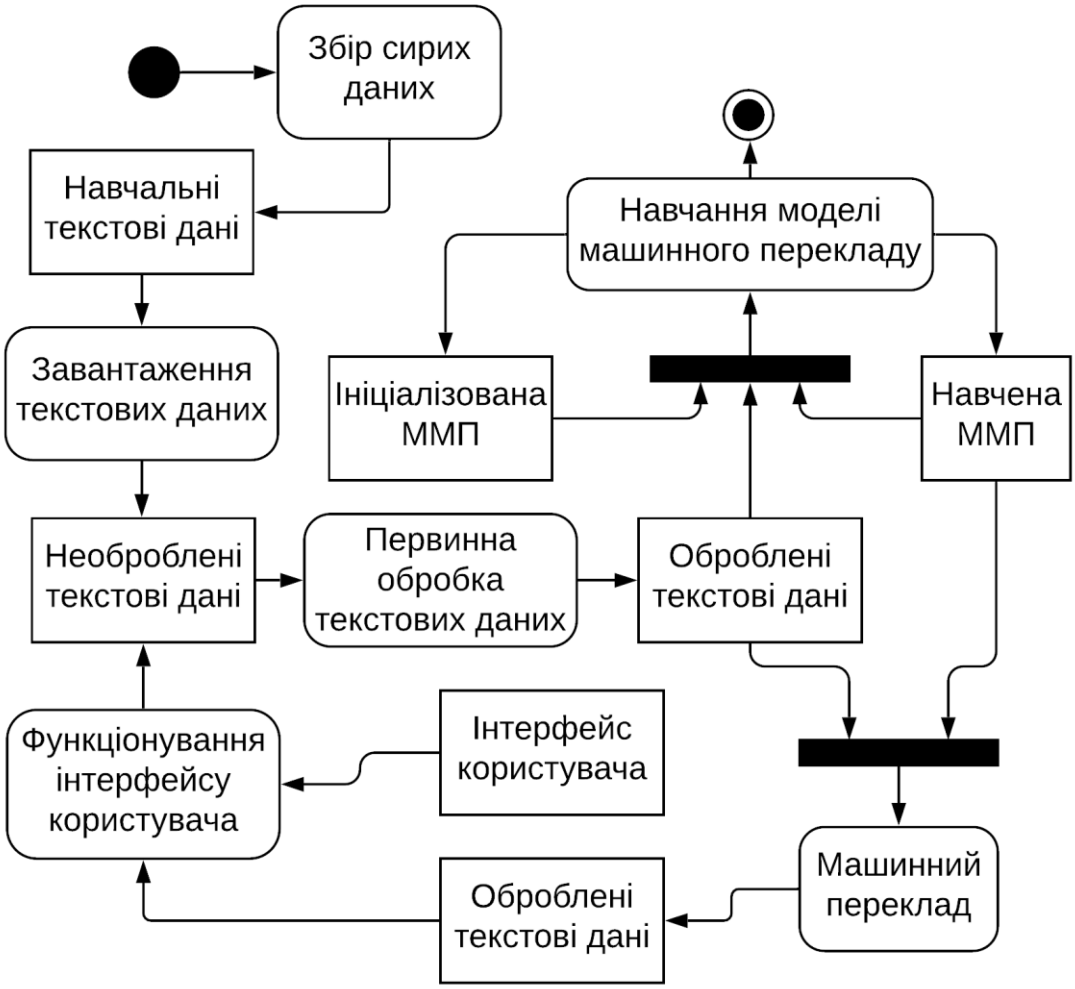


Рис. 3. Модель СНМП. Діаграма діяльності першого рівня в нотації UML

На рис. 4 показано діаграму процесів СНМП із виділенням “водних доріжок”, компонентів (див. рис. 2), отриману на основі вищенаведеної діаграми діяльності рис. 3. Таке представлення дає змогу згрупувати основні процеси системи за відповідними компонентами. Діаграма процесів із водними доріжками є невіддільним складником моделі Еріксона–Пенкера, адже поєднує обидва типи представлень системи та формалізує її діяльність.

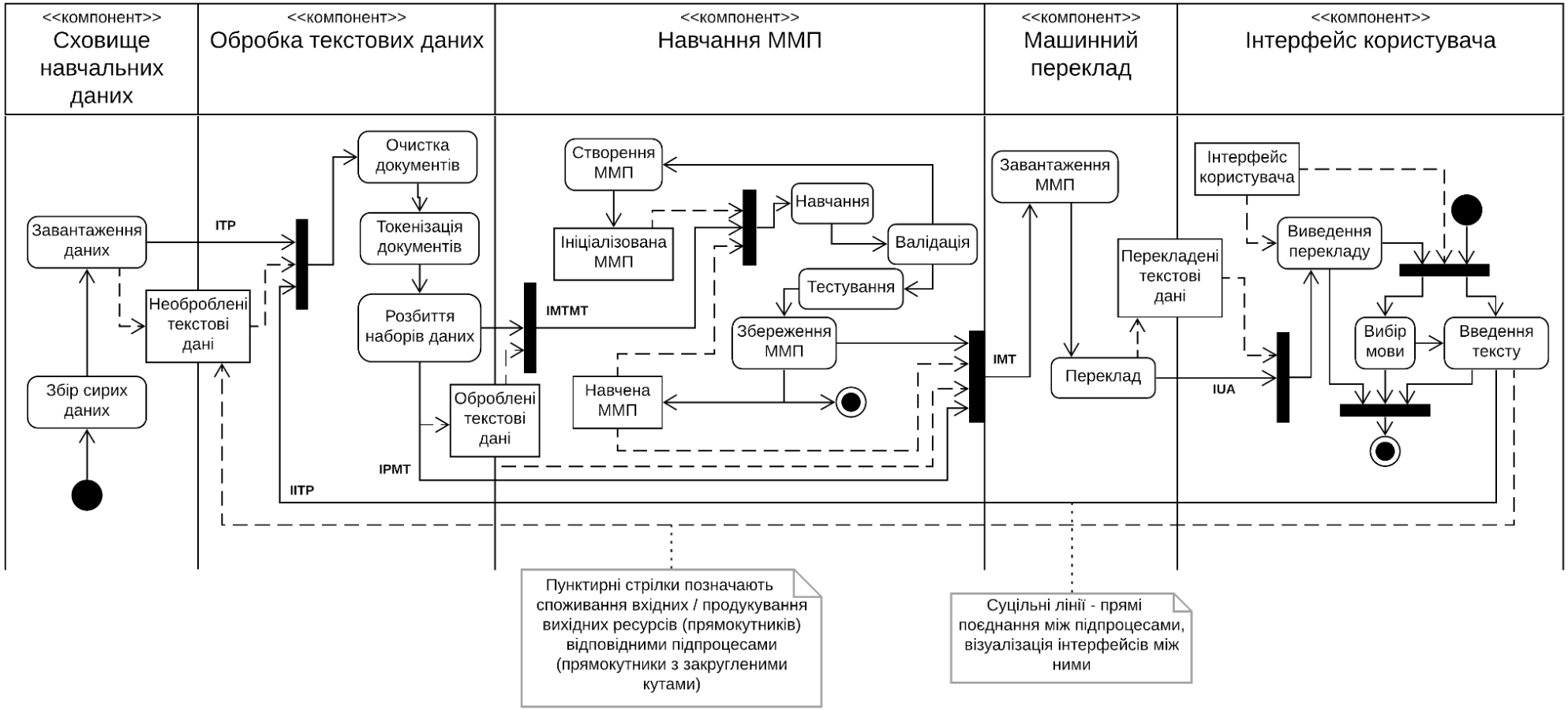


Рис. 4. Модель СНМП. Деталізована діаграма процесів на основі діаграми діяльності з водними доріжками другого рівня в нотації UML

Імплементація компонентів Сховища навчальних й Оброблення текстових даних. Якість навчання будь-якої моделі машинного/глибинного навчання беспосередньо залежить від якості даних, залучених до тренування. Задля високої якості навчального текстового корпусу збирання та оброблення текстів повинні бути систематизовані. Детальний алгоритм збирання та оброблення навчальних текстових такий:

1. Збирання сирих даних:

– збирання/добирання сирих текстових даних із будь-яких відкритих ресурсів (використання відповідних *APIs*, завантаження наборів даних вручну);

– збереження текстових корпусів у табличному форматі: файли з розширенням .csv, .json, файли електронних таблиць *Excel* (.xls, .xlsx) тощо;

– елементарний опис зібраних корпусів: характеристики записів (*features*), кількість записів тощо.

2. Первинне оброблення текстових даних:

– Підпроцес очищення документів:

- зведення даних до стандартних форматів (текстовий формат);
- видалення документів-дублікатів;
- усунення *NaN*, *Null* значень;
- очищення текстів від спеціальних символів, *HTML* тегів;
- декапіталізація слів, аналіз абревіатур;

– під процес токенізації документів: розбиття текстових рядків на список токенів, що в цій імплементації є окремими словами.

3. Розбиття набору даних на тренувальний, валідаційний і тестовий корпуси.

Імплементація компонента Навчання ММП. Процес Навчання ММП:

– побудова навчального словника токенів на основі тренувального корпусу;

– застосування моделі векторного представлення слів (наприклад, *sisg*[11]) з її попереднім тренуванням для показу токенів документів як векторів дійсних чисел (див. п. 3.1);

– створення ініціалізованої ММП початкової архітектури (див. п. 3.2) із використанням підручних програмних бібліотек / завантаження попередньо навченої ММП: останньої версії збереженої ММП або ж попередньо навченої ММП із зовнішніх ресурсів;

– навчання ММП із застосуванням алгоритму оптимізації (наприклад, *Adam* [12]) на тренувальному корпусі $C^{train}C^{train}$ пар паралельних документів ${X,Y}{X,Y}$ оптимізацією цільової функції втрат (наприклад, класичної *перехресної ентропії* (*CE* (*C*ross-*E*ntropy)) [13]) послідовностей токенів YY (дійсний переклад XX) й$\bar{Y}\bar{Y}$;

– валідація для визначення оптимальної архітектури ініціалізованої ММП (тільки під час першої ітерації) та набору гіперпараметрів моделі на валідаційному корпусі $C^{val}C^{val}$ із підрахунком метрики *BLEU score*[14], описаної в п. 5;

– остаточне тестування на тестовому корпусі $C^{test}C^{test}$ навченої на $C^{train}C^{train}$ ММП з оптимальними архітектурою та набором гіпепараметрівіз паралельним підрахунком і візуалізацією визначених метрик (функція втрат і*BLEU score*[14]);

– збереження поточної версії ММП, добір найефективнішої версії моделі з-поміж щойно навченої та попередньо збережених версій для безпосереднього МП запитів користувача.

Визначення критеріїв оцінювання продуктивності системи. Різні версії розроблюваної СНМП (відрізняються різними версіями застосовуваної ММП) порівнюватимуться з точки зору операбельності за часом оброблення (перекладу)запитів користувача. Потенційні кандидати на працездатну систему мають задовольняти BR2 (див. п. 2.2).

Під час проведення під процесів валідації та тестування ММП процесу *Навчання ММП* як метрику оцінювання якості перекладу використовуватимуть *BLEU Score* [14]: отриманий переклад $\hat{Y}$ вхідної текстової послідовності X порівнюють із дійсним відповідним перекладом Y – документом, паралельним до X, наявним у застосовуваному текстовому корпусі. Кожна послідовність складається з токенів $X = \{x_i\}_{i=1}^{I}, Y = \{y_j\}_{j=1}^{J}, \hat{Y} = \{\hat{y}_k\}_{k=1}^{K}$ (де I, J та K – довжини X, Y і $\hat{Y}$ у токенах відповідно). З усіх токенів послідовності $\hat{Y}$ формують і підраховують усі можливі унікальні n-грами токенів довжиною від 1 до N. Функція $Q_{\hat{Y}}(g_n)$ – кількість появ n-грама g_n довжиною n у послідовності $\hat{Y}$. Тоді для всіх послідовностей токенів довжини n вираховують таку величину [14]:

$$p_n = \frac{\sum_{g_n \in \hat{Y}} Q_{\hat{Y}}(g_n)}{\sum_{g_n \in \hat{Y}} Q_Y(g_n)} \forall n = \overline{1, N}. \quad (1)$$

Для всіх довжин n результати обчислень (2.2) комбінують в остаточну *BLEU Score* [14]:

$$BLEU = BP \cdot e^{\frac{1}{N}\sum_{n=1}^{N} \ln(p_n)}, \quad (2)$$

де *BP–BrevityPenalty* (дослівно – штраф стислості) [14]:

$$BP = \begin{cases} 1, & \text{якщо } K > J, \\ 1 - e^{1\frac{J}{K}}, & \text{інакше.} \end{cases} \quad (3)$$

З формул (1–3) очевидним є факт необхідності максимізації *BLEU Score*. Тож найкращу версію перекладача визначатимуть на основі найвищого значення саме цієї метрики.

Верифікація та валідація методу системної інженерії СНМП. На третьому етапі, завдяки запропонованому методу системної інженерії для СНМП, менш ніж за три місяці розроблено першу версію системи двоспрямованого англійсько-українського НМП на основі моделі *sisg* [11] векторного показу слів й архітектури нейронних мереж типу трансформер [10]. Різні версії трансформерів були навчені на корпусах паралельних англійських перекладів ресурсу *OPUS* [15]. Отримана система *EUTM* (*E*nglish-*U*krainian *M*achine *T*ranslator), яказа результатами порівняльного аналізу метрики BLEU Score, поданому в таблиці, щонайменше не поступається за якістю англійсько-українського перекладу загальнодоступному перекладачеві "*Google Translate"*[16].

Жирним шрифтом виділено найкращий показник для двох порівнюваних трансформерів (табл. 1).

***Таблиця* 1.** Порівняння якості перекладу навчених трансформерів із *"Google Translate"*

Версія трансформера (за використаними наборами даних)	**Кількість тренувальних епох**	***BLEU Score*англ-укртрансформера**	***BLEU Score*англ-укр*GT***	***BLEU Score*укр-англтрансформера**	***BLEU Score*укр-англ*GT***
Корпуси *WikiMatrix*[17] і*XLEnt*[18]	503 (4072825 унікальних навчальних документів)	**15.197**	7.772	7.452	**24.165**
Корпус *QED*[19]	5312 (197306 унікальних навчальних документів)	14.122	**14.329**	**20.932**	19.735

Корпус *Tatoeba*[15]	3055 (149038 унікальних навчальних документів)	**23.444**	22.899	**34.037**	10.071

За результатами оцінювання продуктивності систем було встановлено, що розроблена система *EUMT*, розгорнута на віддаленому веб сервер і безплатної платформи *My Binder*, здатна обробляти запити користувача в межах однієї секунди, повністю задовольняючи встановлене бізнес-правило *BR2*.

Повний код розробленої версії системи EUMT опублікований на платформі *GitHub* та доступний за посиланням: https://github.com/EugeneSel/EUMT. Цей репозиторій містить також пряме посилання на веб за стосунок із можливістю його онлайн-тестування.

Висновки

1. Запропоновано метод системної інженерії СНМП, що базується на модифікованому бізнес-профілі Еріксона–Пенкера [1, 5] представлення системи на метарівні, а також міжнародних стандартів процесів Data Science [6] та Data Mining [7], що є основою для алгоритмізації розроблення специфічних для НМП компонентів системи. Досліджено ефективність застосування методу на прикладі розроблення системи двоспрямованого англійсько-українського НМП EUMT і встановлено, що система EUMT щонайменше не поступається за якістю англійсько-українського перекладу популярному перекладачеві "*Google Translate"*.

2. Запропонований метод системної інженерії СНМП спрямовано на розроблення вузькоспеціалізованих СНМП, призначених для первинного перекладу юридичних, медичних, бізнесових й інших важливих текстів, написаних семантично складними мовами, до яких належить й українська мова.

3. Формалізовано розроблення СНМП, що суттєво прискорює й упорядковує імплементацію СНМП і зменшує витрати на її створення.

4. Перспективи подальших досліджень спрямовані на застосування методу системної інженерії СНМП для реалізації СНМП на основі інших математичних моделей МП, формування метрик оцінювання продуктивності та науково обґрунтованих методів верифікації та валідації методу.

References

[1] H.-E. Eriksson and M. Penker, *Businessmodelingwith UML*. NewYork: John Wiley&Sons, 2000, 459 p.

[2] A. Kossiakoff *etal*., *Systems Engineering Principles and Practice*, V.K. Batovrin, Ed. Moscow, Russia: DMKPress, 2014, 624p.

[3] D.K. Hitchins, *Systems Engineering: A 21st Century Systems Methodology*. Wiley, 2007, 528 p.

[4] S. Krymskyi, "Metod," in *Filosofskyi Entsyklopedychnyi Slovnyk*, V.I. Shynkaruk, Ed. Kyiv, Ukraine:Abrys, 2002, 742 p.

[5] P.P. Maslianko and O.S. Maystrenko ,"The system engineering of organizational system in formatization projects,"*KPI Sci. News*, no. 6, pp. 34–42, 2008.

[6] C. O'Neiland R.Schutt, *Doing data science: Straigh ttalk from the front line*. O'ReillyMedia, Inc., 2013, 406 p.

[7] F. Provost and T.Fawcett, *Data science for business: What you need to know about datamining and data-analyticthinking*. O'ReillyMedia, Inc., 2013.

[8] D.Bahdanau*etal*., "Neural machine translation by jointly learning to alignand translate," in *3rd International Conference on Learning Representations*, San Diego, United States, 2014.

[9] J.Gehring*etal*. (2016).*A convolution alencoder model for neural machine translation* [Online]. Available:https://arxiv.org/pdf/1611.02344.pdf

[10] A.Vaswani*etal*., "Attention isall you need," in*31st Conferenceon Neural Information Processing Systems*, Long Beach, CA, USA, 2017.

[11] P.Bojanowski*etal*., "Enriching word vectors with subword information," *Trans. Assoc. Computation. Ling*., vol. 5, pp. 135–146, 2017.

[12] D.P. KingmaandJ.L. Ba. (2014). *Adam: A method for stochastic optimization*[Online]. Available:https://arxiv.org/pdf/1412.6980.pdf?source=post_page

[13] C.Szegedy*etal*.(2016). *Rethinking the inception architecture for computer vision* [Online]. Available:https://arxiv.org/pdf/1512.00567.pdf

[14] K.Papineni*etal*.“Bleu: A method for automaticev aluation of machine translation,”in *Proc. 40th Annual Meeting of the Association for Computational Linguistics(ACL)*, Philadelphia, 2002, pp. 311–318.

[15] J.Tiedemann.(2012). *Paralleldata, tools and interfacesin OPUS* [Online]. Available:http://www.lrec-conf.org/proceedings/lrec2012/pdf/463_Paper.pdf

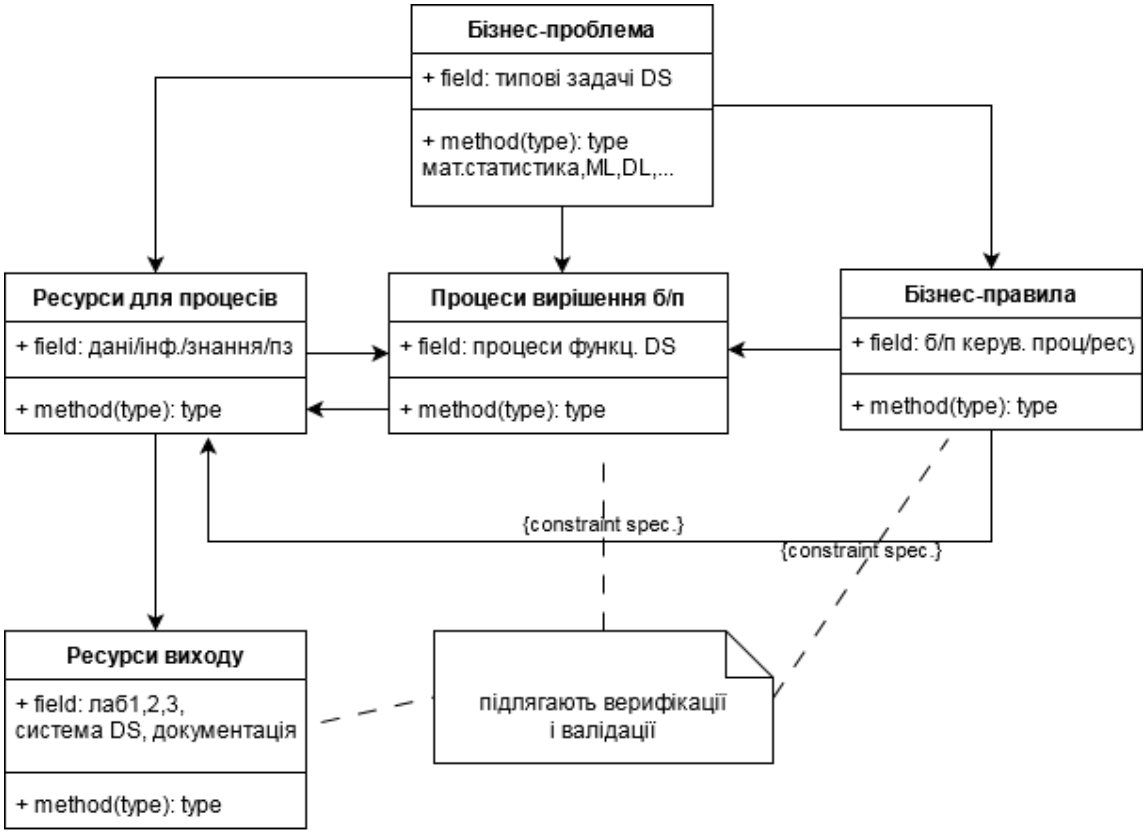
[16] M.Johnson*etal*.(2017). *Google’smultilingual neural machine translation system: Enabling zero-shottranslation* [Online]. Available:https://arxiv.org/pdf/1611.04558.pdf

[17] S. Holger*etal*. (2019). *Wiki Matrix: Mining 135M parallel sentencesin 1620 languag epairs from Wikipedia* [Online]. Available:https://arxiv.org/pdf/1907.05791.pdf

[18] A.El-Kishky*etal*. (2021). *XLEnt: Mining Cross-lingual Entitieswith Lexical-Semantic-Phonetic Word Alignment*[Online]. Available:http://data.statmt.org/xlent/elkishky_XLEnt.pdf

19. A. Abdelali*etal*., “The AMARA Corpus: Building parallell anguagere sources for the educational domain,” in *Proc. 9th International Conferenceon Language Resourcesand Evaluation (LREC'14)*, Reykjavik, Iceland, 2014, p. 1856–1862.

Спрощена версія бізне-профіля Еріксона-Пенкера



Спрощений бізнес-профіль Еріксона-Пенкера. Діаграма класів в нотації UML

7. Лабораторні роботи № 1, 2, 3.

Формат, зміст і завдання лабораторних робіт.

Назва документа Лаб№1 DSKM-XX ПІБ повністю

Лабораторна робота №1 Інженерія систем DS.

Назва системи DS

КМ-ХХ, ПІБ повністю.

Розуміння бізнесу - розуміння даних. Формалізація постановки задачі інженерії систем DS.

Завдання.

Розуміння бізнесу — розуміння даних:

– визначення предметної області для дослідження і попередня постановка задачі інженерії системи DS в рамках конкретного контексту з обраної предметної області; (Див приклад системної інженерії СНМП, типові задачі моделювання).

Бізнес-профіль Еріксона-Пенкера інженерії системи Data Science. Сутності бізнес-профіля Еріксона-Пенкера для інженерії системи Data Science:

– **бізнес-проблема і мета** вирішення бізнес-проблеми (типова задача/задачі Data Dcience);

– **процеси/функції** вирішення бізнес-проблеми (перелік процесів і алгоритмів статистичного аналізу, ML, DL для досягнення мети і вирішення бізнес-проблеми, Міжгалузевий стандарт процесу майнингу даних CRISP-DM, Shearer, 2000);

– **ресурси** для реалізації бізнес-процесів і вирішення бізнес-проблеми, математичні і інструментальні засоби інженерії системи Data Science (методи і моделі статистичного аналізу, ML, DL,, інструменти Scikit – learn *https://scikit-learn.org/stable/* , *https://www.kaggle.com/* , бібліотека *pandas* ,та інші);

– **бізнес-правила** для організації і упорядкування застосування ресурсів для реалізації бізнес-процесів вирішення задачі Data Science та вирішення бізнес-проблеми.

Модель системи Data Science. Діаграма класів в нотації UML, Діаграма компонентів в нотації UML, Діаграма діяльності в нотації UML.

– збір/вибір сирих даних з будь-яких відкритих ресурсів (використання відповідних *API*s, завантаження наборів даних вручну);

– збереження ресурсів даних які потрібні для вирішення конкретної задачі, в певному форматі (наприклад, у випадку *API*s): табличні дані CSV, або інший формат з роздільниками, файли електронних таблиць Excel (.xls, .xslsx), OpenDocument (.ods), тощо (бібліотека *pandas* надає широкий функціонал керування даними в табличному вигляді — *dataframe*);

– візуалізація (елементарний опис зібраного набору даних: характеристики записів (features), кількість записів і т.д., наприклад, *pandas* функція *info*) і документування результатів/документування класів бізнес-профіля Еріксона-Пенкера.

1.1 Підготовка даних:

– первинна очистка даних (*pandas*):
  - приведення даних до стандартних форматів (числові, текстові формати, усунення *objects*);
  - видалення записів-дублікатів;
  - усунення *NaN, Null* значень (наприклад, заміна на середні значення відповідних характеристик або видалення записів з *NaN, Null* значеннями);

– розвідувальний аналіз даних — *EDA, Exploratory Data Analysis* (*matplotlib*, *seaborn*):
  - застосування описової статистики — розрахунок середніх арифметичних, дисперсій, мод, медіан усіх характеристик (наприклад, *pandas* функція *describe*);

- проведення кореляційного та причинно-наслідкового аналізів (наприклад, див. *pandas* функцію *corr*, *seaborn* функції *heatmap*, *pairplot*, *jointplot*);
- тощо;

– конструювання ознак (*feature engineering*) на основі проведеного EDA (за необхідності):

- видалення зайвих характеристик;
- поєднання/композиція (наприклад, визначення нової характеристики x_3 як добуток існуючих характеристик x_1 та x_2, тобто $x_3 = x_1 \cdot x_2$) характеристик у нові, нелінійні;
- кодування категоріальних (*categorical*) характеристик (наприклад, *one-hot encoding*, *pandas* функція *get_dummies*, звернути увагу на тип даних *categorical*);

– розбиття набору даних на тренувальний, валідаційний (за необхідності) та тестовий набори даних (наприклад, *scikit-learn* функція *train_test_split*);

– візуалізація і документування результатів/документування класів бізнес-профіля Еріксона-Пенкера.

1.2 Верифікація і валідація.

1.3 Висновки.

Лабораторна робота №2 Системи DS.

Назва системи DS

КМ-ХХ, ПІБ повністю.

Моделювання.

Завдання.

1. Перелік існуючих потенційних методів розв’язання поставленої задачі, короткий опис обраного методу моделювання (статистика, ML), його математичного забезпечення.

2. Формалізація цільової функції оптимізації (функція втрат — *loss function*) в залежності від типу задачі моделювання (наприклад, *MSE* для регресії, (*Binary*) *Cross Entropy* для класифікації тощо). Визначення метрик оцінки ефективності моделі (наприклад, *accuracy* для класифікації).

3. Імплементація обраного методу моделювання:

– створення моделі з використанням підручних бібліотек: наприклад, *scikit-learn*, *PyTorch*, *TensorFlow* (з його *API* високого рівня *Keras*);

– за необхідності — стандартизація числових даних (*feature scaling*). *УВАГА*: середні арифметичні та стандартні відхилення характеристик визначаються на тренувальних даних та застосовуються для стандартизації всіх трьох наборів даних;

– навчання моделі (зазвичай, функція *fit* моделей *scikit-learn*, *PyTorch*, *TensorFlow*, *Keras*) на тренувальному наборі даних з паралельним підрахунком та візуалізацією визначених метрик;

– за необхідності — проведення валідації гіперпараметрів моделі на валідаційному наборі даних з паралельним підрахунком визначених метрик, оптимізація гіперпараметрів (*hyperparameter tuning*, вручну, систематично — *Grid Search*, автоматично з використанням таких бібліотек, як *scikit-optimize*, *optuna* та ін.), побудова навчальних кривих (*learning curves*) для метрик, функція втрат для тренувальних та валідаційних даних;

– остаточне (**одноразове**) тестування навченої моделі (зазвичай, функції *predict* чи *evaluate* моделей *scikit-learn*, *PyTorch*, *TensorFlow*, *Keras*) на тестових даних з паралельним підрахунком та візуалізацією визначених метрик

– інтерпретація отриманих результатів: опис ефективності обраного методу в контексті вирішення поставленої задачі, його можливі недоліки/переваги в рамках конкретного застосування, опис впливу попередньо проведених маніпуляцій над даними/розробки моделі (п. 1.2 Підготовка даних лабораторної 1, власне п. 3 даної лабораторної — імплементація моделі) на отримані результати, приведення потенційних модифікацій для їхнього покращення;

– візуалізація і документування результатів/документування класів бізнес-профіля Еріксона-Пенкера;

4. Верифікація і валідація.
5. Висновки.

Назва документа Лаб№3 DSKM-XX ПІБ повністю

Лабораторна робота №3 Імплементація Системи DS.

Назва системи DS

КМ-XX, ПІБ повністю

Система DS. Представлення системи DS, систематизація результатів та документування.

Завдання.

1. Постановка задачі інженерії системи DS, бізнес-профіль Еріксона-Пенкера.

2. Структурне і динамічне представлення системи DS.

Формалізовані бізнес-правила вирішення завдань.

Документація:

- Модель системи DS. Діаграма класів в нотації UML. Можна зробити виноску змісту класу у вигляді окремого класу (коментаря) з повним переліком змісту (начинки) цього класу;
- Модель системи DS. Діаграма компонентів в нотації UML. Можна зробити виноску змісту компонента у вигляді окремого компонента (коментаря) з повним переліком змісту (начинки) цього компонента;
- Модель системи DS. Діаграма діяльності в нотації UML.
- Математичне і програмне забезпечення у відповідності до змісту компонента Ресурси (ресурси входу, резурси забезпечення діяльності системи DS, ресурси виходу, тобто результати);
- Графіки, таблиці, діаграми і висновки попередньо отриманих результатів лабораторних робіт 1 і 2 для формування висновків щодо представлення системи DS лабораторної роботи №3;

3. Верифікація і валідація системи DS у цілому.

4. Висновки щодо прийнятих припущень і формалізованих бізнес-правил, повноти вибору інструментів і вирішення завдань моделювання, представлення системи DS і повноти документації на систему DS.

Можливі формати звітування лабораторних робіт №№1, 2, 3

(для обговорення)

1. **Оформлення звіту в форматі .docx з дотриманням ДСТУ 3008-2015 (титульна сторінка, постановка задачі, основна частина, висновки), наповнення розділів — згідно із завданнями на лабораторні, + лістинг коду в додатку.**

2. Повноцінна розробка в локальному .ipynb ноутбуці з додаванням *markdown* клітин (див. статтю для початківців) з детальним описом проведених робіт (згідно із завданнями на лабораторні), описом математичного апарату (додавання формул до *markdown* клітин), відповідною структурою (нумерація розділів, підрозділів, виділення тексту і т.д.). Збереження ноутбука в .pdf форматі для надсилання викладачеві.

3. Повноцінна розробка в *Google Colab* ноутбуці на власному *Google Drive* з вищевказаними вимогами до оформлення ноутбука (див. п.2). Надсилання викладачеві посилання на готовий ноутбук з можливістю коментувати.