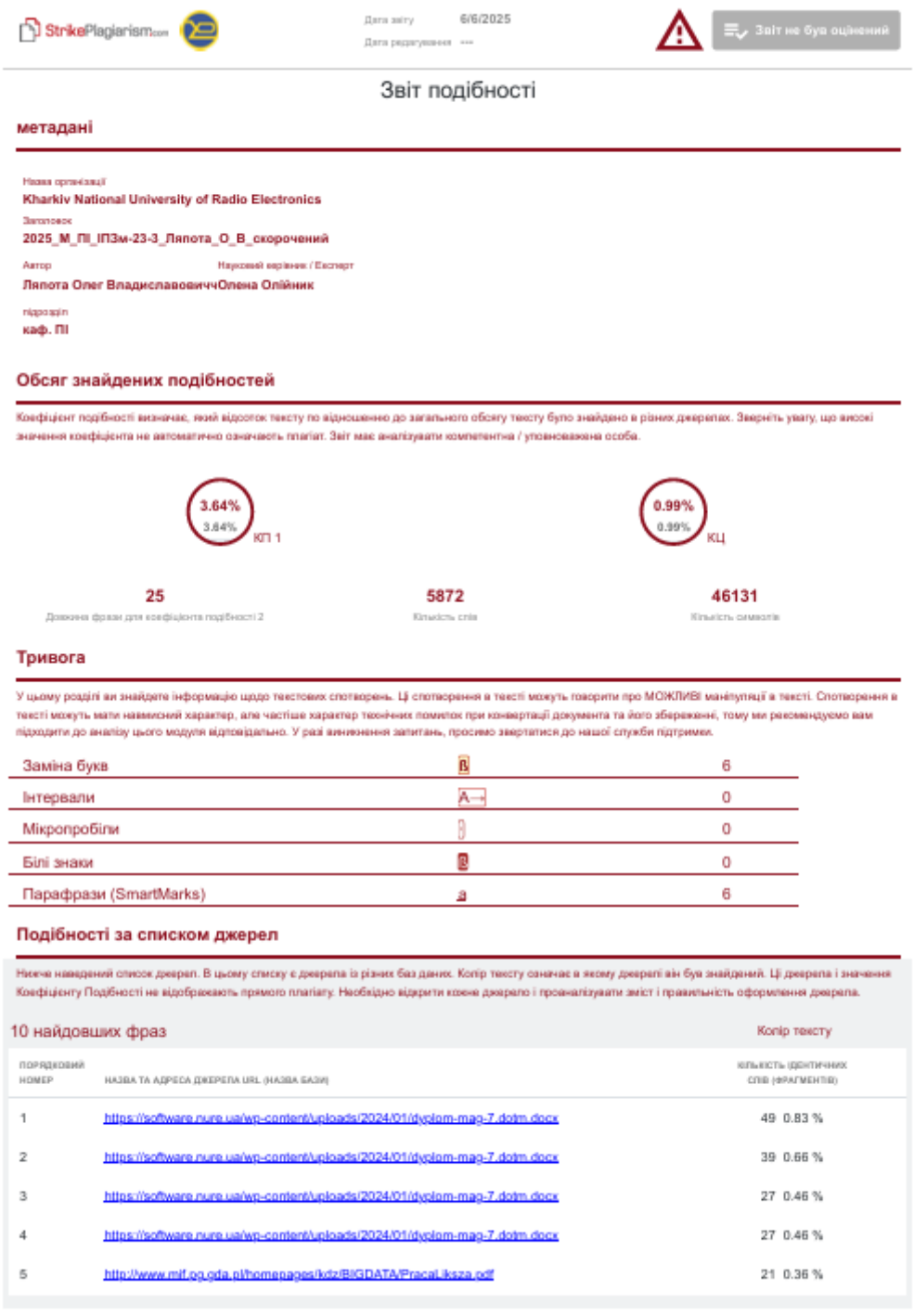
ДОДАТОК А

Звіт результатів перевірки на унікальність тексту в базі ХНУРЕ



StrikePlagiarism.com

Дата звіту 6/6/2025
Дата редагування ---

Звіт не був оцінений

Звіт подібності

метадані

Назва організації
Kharkiv National University of Radio Electronics
Заголовок
2025_M_ПІ_ІПЗм-23-3_Ляпота_О_В_скорочений
Автор
Ляпота Олег Владиславович
Науковий керівник / Експерт
Олена Олійник
підрозділ
каф. ПІ

Обсяг знайдених подібностей

Коефіцієнт подібності визначає, який відсоток тексту по відношенню до загального обсягу тексту було знайдено в різних джерелах. Зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.

3.64% 3.64% КП 1

0.99% 0.99% КЦ

25
Довжина фрази для коефіцієнта подібності 2

5872
Кількість слів

46131
Кількість символів

Тривога

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про МОЖЛИВІ маніпуляції в тексті. Спотворення в тексті можуть мати навмисний характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

Заміна букв	в	6
Інтервали	A→	0
Мікропробіли	¶	0
Білі знаки	␣	0
Парафрази (SmartMarks)	a	6

Подібності за списком джерел

Нижче наведений список джерел. В цьому списку є джерела із різних баз даних. Колір тексту означає в якому джерелі він був знайдений. Ці джерела і значення Коефіцієнту Подібності не відображають прямого плагіату. Необхідно відкрити кожне джерело і проаналізувати зміст і правильність оформлення джерела.

10 найдовших фраз — Колір тексту

ПОРЯДКОВИЙ НОМЕР	НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ)	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	https://software.nure.ua/wp-content/uploads/2024/01/dyplom-mag-7.dotm.docx	49 0.83 %
2	https://software.nure.ua/wp-content/uploads/2024/01/dyplom-mag-7.dotm.docx	39 0.66 %
3	https://software.nure.ua/wp-content/uploads/2024/01/dyplom-mag-7.dotm.docx	27 0.46 %
4	https://software.nure.ua/wp-content/uploads/2024/01/dyplom-mag-7.dotm.docx	27 0.46 %
5	http://www.mif.pg.gda.pl/homepages/kdz/BIGDATA/PracaLiksza.pdf	21 0.36 %

6	Курсова Машинне навчання: Визначення наявності діабету [redacted] 12/21/2024 Lutsk National Technical University course papers (Lutsk National Technical University course papers)	15 0.26 %
7	https://www.frontiersin.org/articles/10.3389/fcell.2021.659941/full	13 0.22 %
8	https://dukt.edu.ua/repozitorii/ai/2024/%D0%9E%D1%81%D1%82%D1%80%D0%B5%D0%BD%D1%81%D1%8C%D0%BA%D0%B8%D0%B9%20%D0%B4%D0%B8%D0%BF%D0%BB%D0%BE%D0%BC%20%D0%B1%D0%B0%D0%BA%D0%B0%D0%BB%D0%B0%D0%B2%D1%80%D0%B0%20%D0%A8%D0%86%D0%94-41.pdf	11 0.19 %
9	The Best Answers? Think Twice: Online Detection of Commercial Campaigns in the CQA Forums Venkatesh Srinivasan,Cheng Chen,Kesav Bharadwaj R,Kui Wu;	6 0.10 %
10	https://software.nure.ua/wp-content/uploads/2024/01/dyplom-mag-7.dotm.docx	6 0.10 %

з бази даних RefBooks (0.10 %)

ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (#ФРАГМЕНТІВ)
джерело: https://arxiv.org/		
1	The Best Answers? Think Twice: Online Detection of Commercial Campaigns in the CQA Forums Venkatesh Srinivasan,Cheng Chen,Kesav Bharadwaj R,Kui Wu;	6 (1) 0.10 %

з домашньої бази даних (0.00 %)

ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (#ФРАГМЕНТІВ)

з програми обміну базами даних (0.26 %)

ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (#ФРАГМЕНТІВ)
1	Курсова Машинне навчання: Визначення наявності діабету [redacted] 12/21/2024 Lutsk National Technical University course papers (Lutsk National Technical University course papers)	15 (1) 0.26 %

з Інтернету (3.29 %)

ПОРЯДКОВИЙ НОМЕР	ДЖЕРЕЛО URL	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (#ФРАГМЕНТІВ)
1	**https://software.nure.ua/wp-content/uploads/2024/01/dyplom-mag-7.dotm.docx**	**148 (5) 2.52 %**
2	http://www.mif.pg.gda.pl/homepages/kdz/PliGDATA/PracaLiksza.pdf	21 (1) 0.36 %
3	https://www.frontiersin.org/articles/10.3389/fcell.2021.659941/full	13 (1) 0.22 %
4	https://dukt.edu.ua/repozitorii/ai/2024/%D0%9E%D1%81%D1%82%D1%80%D0%B5%D0%BD%D1%81%D1%8C%D0%BA%D0%B8%D0%B9%20%D0%B4%D0%B8%D0%BF%D0%BB%D0%BE%D0%BC%20%D0%B1%D0%B0%D0%BA%D0%B0%D0%BB%D0%B0%D0%B2%D1%80%D0%B0%20%D0%A8%D0%86%D0%94-41.pdf	11 (1) 0.19 %

Список прийнятих фрагментів (немає прийнятих фрагментів)

ПОРЯДКОВИЙ НОМЕР	ЗМІСТ	КІЛЬКІСТЬ ОДНАКОВИХ СЛІВ (#ФРАГМЕНТІВ)

2
ПЕРЕЛІК СКОРОЧЕНЬ
AI – Artificial Intelligence
API – Application Programming Interface

ДОДАТОК Б

Слайди презентації

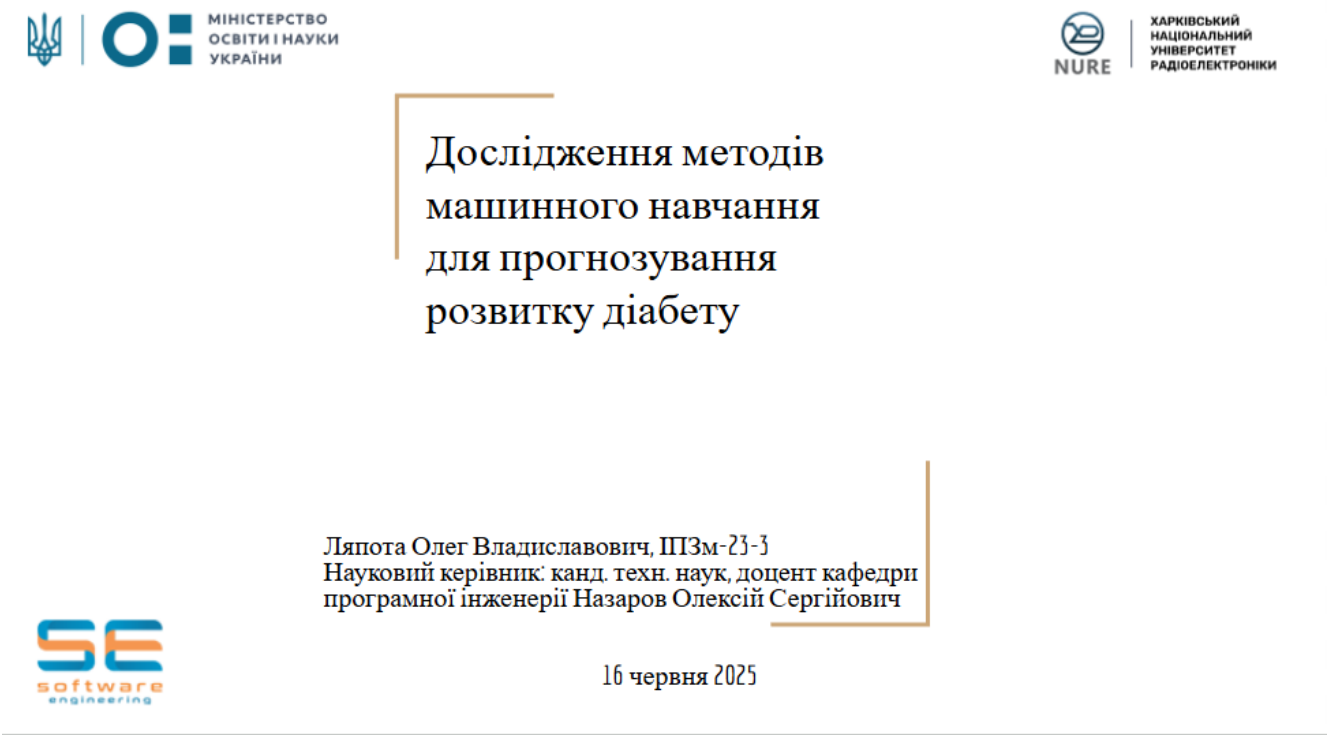


Рисунок Б.1 – Титульний слайд

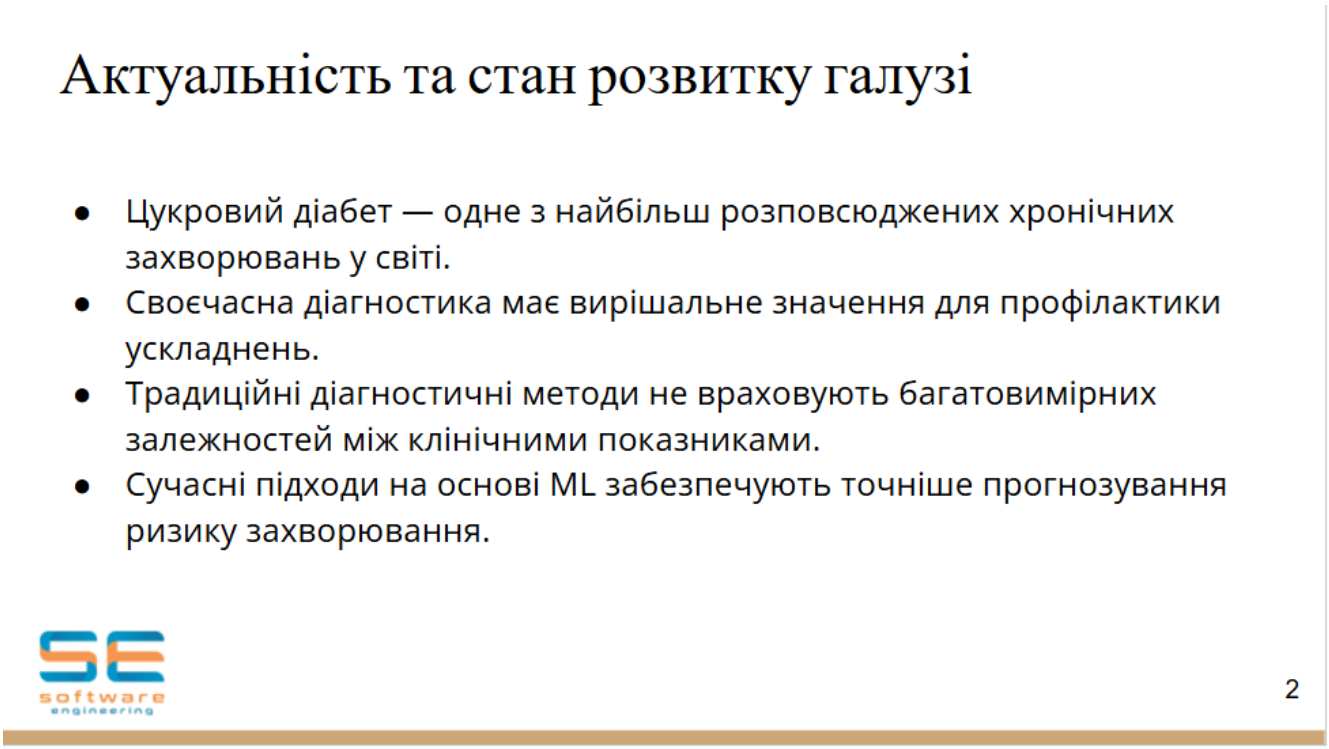


Рисунок Б.2 – Актуальність та стан розвитку галузі

Дослідження

Напрямок дослідження: Розробка та експериментальна оцінка інтелектуальної системи для прогнозування ризику розвитку діабету на основі алгоритмів машинного навчання з подальшою інтеграцією у REST-сервіс.

Об'єкт дослідження: Процес медичної діагностики ризику розвитку цукрового діабету за допомогою аналітичної обробки клінічних даних.

SE software engineering

3

Рисунок Б.3 – Напрямок та об'єкт дослідження

Огляд літератури (аналогів)

- Alghamdi M, Al-Mallah M, Keteyian S, Brawner C, Ehrman J, et al. (2017) Predicting diabetes mellitus using SMOTE and ensemble machine learning approach: The Henry Ford ExercIse Testing (FIT) project. PLOS ONE 12(7): *розглядає ансамблеві методи та SMOTE для покращення класифікації діабету*
- Dagliati A, Marini S, Sacchi L, et al. Machine Learning Methods to Predict Diabetes Complications. Journal of Diabetes Science and Technology. 2018;12(2):295-302 *розглядає інтеграцію моделей ML у клінічні системи.*
- El-Sofany, Hosam, El-Seoud, Samir A., Karam, Omar H., Abd El-Latif, Yasser M., Taj-Eddin, Islam A. T. F., A Proposed Technique Using Machine Learning for the Prediction of Diabetes Disease through a Mobile App, International Journal of Intelligent Systems, 2024, 6688934, 13 pages, 2024 розглядає застосування мобільних ML-додатків для медичної діагностики.

SE software engineering

4

Рисунок Б.4 – Огляд літератури та рішень аналогі

Постановка задачі

Традиційні методи діагностики діабету є переважно реактивними та неефективні для раннього виявлення захворювання, у результаті виникає потреба у створенні системи, здатної інтелектуально оцінювати ризик розвитку діабету на основі багатовимірних медичних даних.

Додатково, при роботі з медичними даними виникають проблеми з незбалансованістю даних та необхідність в інтерпретованості результатів

В результаті, для вирішення наведених проблем необхідно створити точну, гнучку та клінічно придатну систему підтримки медичних рішень на основі машинного навчання.

Рисунок Б.5 – Постановка задачі

Методи дослідження

Дослідження проводиться з використанням методів ML:

- Decision Tree на базі інформаційного зиску: $IG(S, A) = Entropy(S) - \sum_{v \in Values(A)} |S_v||S| \cdot Entropy(S_v)$
- Ансамблеві розширення Decison Tree: Random Forest і XGBoost
- Logistic Regression: $h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$ $J(\theta) = -\frac{1}{m} \sum_{i=1}^{m} \left[y^{(i)} \log\left(h_\theta\left(x^{(i)}\right)\right) + \left(1 - y^{(i)}\right) \log\left(1 - h_\theta\left(x^{(i)}\right)\right) \right]$
- CNN
- MLP

Рисунок Б.6 – Методи дослідження

Інструментарій та технології

- Мова програмування: Python 3.11
- ML-бібліотеки: scikit-learn, xgboost, Keras/TensorFlow
- REST API: Flask + JWT-авторизація
- Підключення до бази даних: SQLite, SQLAlchemy ORM
- Обробка даних: pandas, numpy
- Візуалізація: matplotlib, seaborn
- Система зберігання моделей: joblib, .h5 для нейромереж
- Реляційна БД: MySQL
- Не-реляційна БД: MongoDB

Рисунок Б.7 – Інструменти та технології

Архітектура система для проведення експериментального дослідження

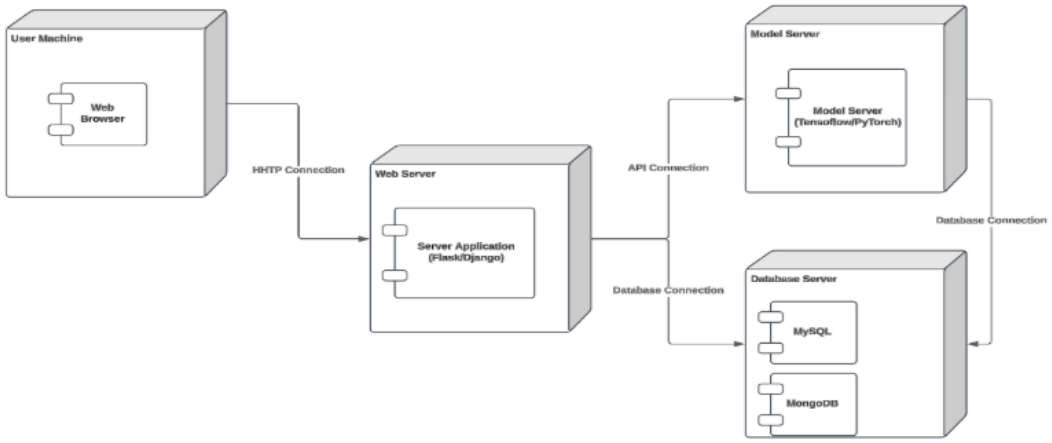


Рисунок Б.8 – Архітектура системи

Архітектура системи для проведення експериментального дослідження

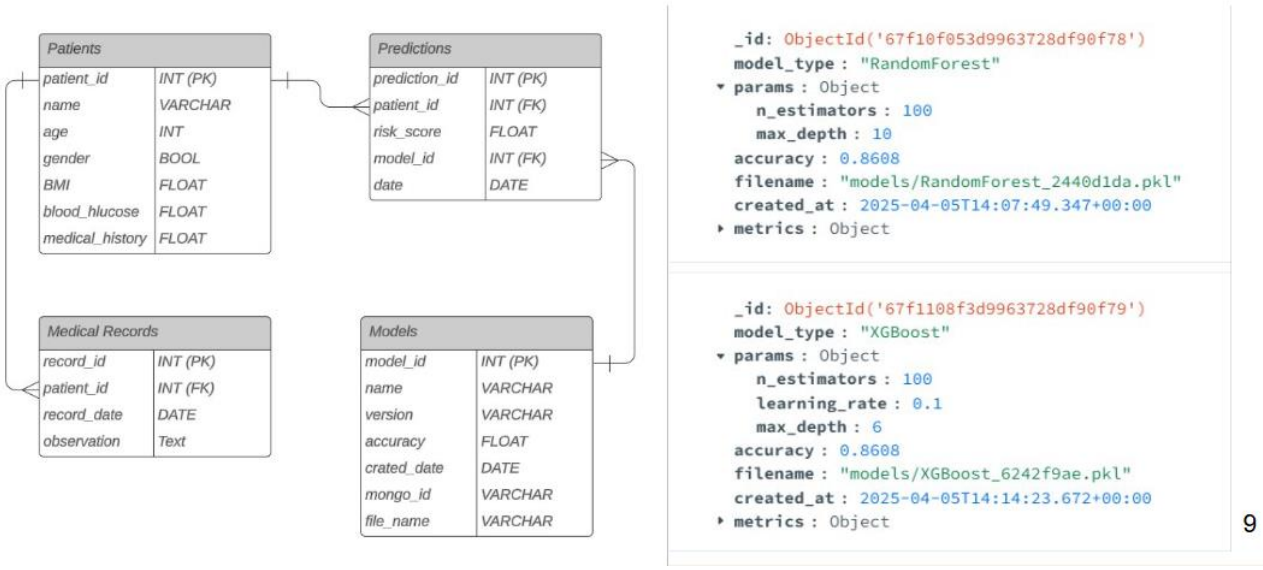


Рисунок Б.9 – Архітектура баз даних

Зміст проведеного експерименту

- Вхідні дані: Pima Indians Diabetes Dataset: 768 записів, 8 клінічних параметрів, об’єкт прогнозування;
- Критерії оцінки: Accuracy, Precision, Recall, F1 Score, ROC AUC;
- Послідовність:
  - Завантаження та очищення даних.
  - Генерація нових ознак.
  - Балансування класів.
  - Навчання моделей.
  - Оцінка метрик.
  - Збереження результатів через REST API.
- Результати: розраховуються на валідаційний вибірці

SE software engineering

10

Рисунок Б.10 – Зміст проведеного експерименту

Кількісні результати експерименту

	Accuracy	Precision	Recall	F1 Score	ROC AUC
Logistic Regression	84.81%	83.33%	87.50%	85.37%	91.54%
Decision Tree	79.75%	74.00%	92.50%	82.22%	84.71%
Random Forest	86.08%	83.72%	90.00%	86.75%	94.78%
XGBoost	86.08%	82.22%	92.50%	87.06%	94.17%
MLP	82.28%	80.95%	85.00%	82.93%	90.96%
CNN	83.54%	81.40%	87.50%	84.34%	90.38%

11

Рисунок Б.11 – Кількісні результати експерименту

Якісні результати експерименту

Важливість критерію	Назва критерію	Розрахунок коефіцієнту
1	ROC AUC	$\frac{1}{15} = 0.067$
2	Recal	$\frac{2}{15} = 0.133$
3	Precision	$\frac{3}{15} = 0.2$
4	Accuracy	$\frac{4}{15} = 0.267$
5	F1 Score	$\frac{5}{15} = 0.333$

Logistic Regresion — 85.48;

Random Forest — 86.91;

XGBoost — 87.04;

CNN — 84.33;

12

Рисунок Б.12 – Якісні результати експерименту

Аналіз отриманих результатів

Досягнуто основну мету — розробка та експериментальне підтвердження ефективної системи прогнозування діабету на основі ML.

XGBoost і Random Forest демонструють найкращі метрики (F1 > 87%, AUC > 94%). Генерація нових ознак покращила продуктивність усіх моделей.

Підтверджено доцільність використання ансамблевих методів у медичному прогнозуванні. Моделі мають високу Recall, що критично для задач попередньої діагностики. Висока F1 Score свідчить про узгодженість передбачень.

software engineering

13

Рисунок Б.13 – Аналіз отриманих результатів

Публікація результатів

software engineering

14

Рисунок Б.14 – Публікація результатів

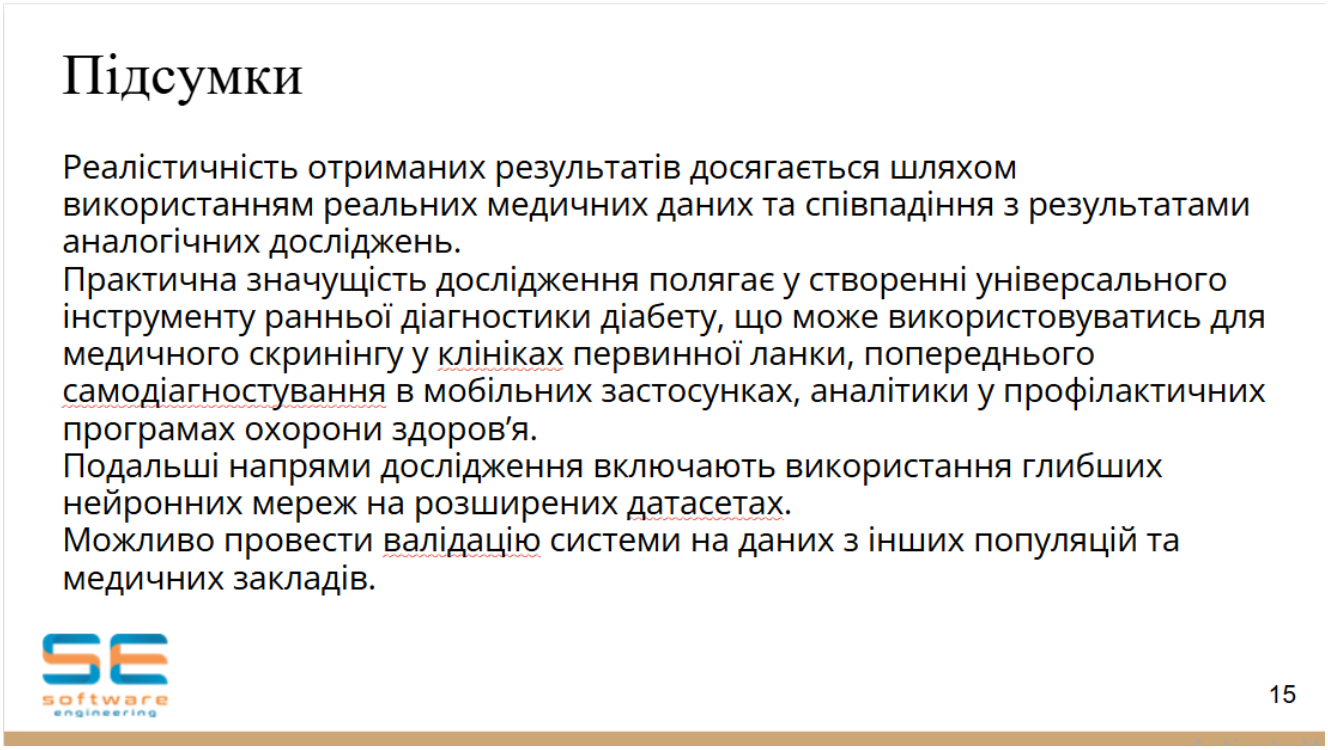


Рисунок Б.15 – Підсумки

ДОДАТОК В

Приклади коду програм

```
     def generate_version_hash(model_type, params):
    data = json.dumps({"model_type": model_type, "params": params},
sort_keys=True)
    return hashlib.md5(data.encode()).hexdigest()[:8]


     def preprocess_features(df):
    for col in df.columns:
        df[col] = pd.to_numeric(df[col], errors='coerce')
    df["Glucose_BMI_ratio"] = df["Glucose"] / df["BMI"]
    df["Is_obese"] = (df["BMI"] >= 30).astype(int)
    df["Age_group"] = pd.cut(df["Age"], bins=[0, 30, 50, 120], labels=[0,
1, 2])
    df["HbA1c_level"] = pd.cut(df["Glucose"], bins=[0, 100, 125, 300],
labels=[0, 1, 2])
    return df


     model.fit(X_train, y_train)
metrics = evaluate_model(model, X_val, y_val)

version_hash = generate_version_hash(model_type, params)
filename = f"models/{model_type}_{version_hash}.pkl"
joblib.dump(model, filename)

mongo_id = save_model_to_mongo(model_type, params, filename,
metrics["accuracy"], metrics)
model_id = save_model_metadata(model_type, version_hash,
metrics["accuracy"], mongo_id, filename)

return {
    "status": " Model trained",
    "model_type": model_type,
    "accuracy": float(metrics["accuracy"]),
    "model_id": model_id,
    "filename": filename,
    "params": params
}


        X_train_np = np.expand_dims(X_train.values, axis=-1)
    y_train_np = y_train.values

    X_val_np = np.expand_dims(X_val.values, axis=-1)
    y_val_np = y_val.values

    model = Sequential([
        Input(shape=(X_train_np.shape[1], 1)),
        Conv1D(filters=params.get("filters", 32), kernel_size=3,
activation='relu'),
        Flatten(),
```

```
        Dense(params.get("dense_units", 64), activation='relu'),
        Dropout(params.get("dropout", 0.3)),
        Dense(1, activation='sigmoid')
    ])

    model.compile(
        optimizer=Adam(learning_rate=params.get("learning_rate", 0.001)),
        loss='binary_crossentropy',
        metrics=['accuracy']
    )

    model.fit(
        X_train_np, y_train_np,
        validation_data=(X_val_np, y_val_np),
        epochs=params.get("epochs", 20),
        batch_size=params.get("batch_size", 32),
        verbose=0
    )

    metrics = evaluate_model(model, X_val_np, y_val_np, is_cnn=True)
    version_hash = generate_version_hash(model_type, params)
    filename = f"models/{model_type}_{version_hash}.keras"
    model.save(filename)

    mongo_id = save_model_to_mongo(model_type, params, filename,
metrics["accuracy"], metrics)
    model_id = save_model_metadata(model_type, version_hash,
metrics["accuracy"], mongo_id, filename)

    return {
        "status": " CNN trained",
        "model_type": model_type,
        "accuracy": float(metrics["accuracy"]),
        "model_id": model_id,
        "filename": filename,
        "params": params
    }

else:
    return {"error": f"Unsupported model_type: {model_type}"}


def predict_risk(df, model_id=None):
    if model_id:
        session = Session()
        model_entry = session.query(Model).filter(Model.model_id ==
model_id).first()
        session.close()


        if not model_entry:
            return {"error": f"Model with ID {model_id} not found"}
```

```
        filename = model_entry.filename
        if not filename:
            return {"error": f"No filename associated with model ID
{model_id}"}
    else:
        filename = "models/model.pkl"

    if not os.path.exists(filename):
        return {"error": f"Model file {filename} not found"}

    if filename.endswith(".h5") or filename.endswith(".keras"):
        model = load_model(filename)
        df = preprocess_features(df)
        X = df[FEATURES].values
        X = np.expand_dims(X, axis=-1)  # (samples, features, 1)
        prob = model.predict(X)[0][0]
        return {"risk_score": float(prob), "model_file": filename,
"model_type": "CNN"}

    else:
        model = joblib.load(filename)
        df = preprocess_features(df)
        X = df[FEATURES]
        prob = model.predict_proba(X)[:, 1]
        return {"risk_score": float(prob[0]), "model_file": filename}


     def get_model_metrics(model_id):
    session = Session()
    model_entry = session.query(Model).filter(Model.model_id ==
model_id).first()
    session.close()

    if not model_entry:
        return {"error": f"Model with ID {model_id} not found"}

    filename = model_entry.filename
    mongo_id = model_entry.mongo_id

    if not filename or not os.path.exists(filename):
        return {"error": f"Model file {filename} not found"}

    config = get_model_config(mongo_id)
    if config and "metrics" in config:
        return config["metrics"]
```

```
df = load_dataset("validation_data")
X = df[FEATURES]
y = df["Outcome"]

if filename.endswith(".h5") or filename.endswith(".keras"):
    model = load_model(filename)
    X_np = np.expand_dims(X.values, axis=-1)
    metrics = evaluate_model(model, X_np, y, is_cnn=True)
else:
    model = joblib.load(filename)
    metrics = evaluate_model(model, X, y)

update_model_metrics_in_mongo(mongo_id, metrics)

return metrics


 def evaluate_model(model, X, y, is_cnn=False):
if is_cnn:
    y_pred = model.predict(X).flatten()
    y_pred_label = (y_pred >= 0.5).astype(int)
else:
    y_pred = model.predict_proba(X)[:, 1]
    y_pred_label = (y_pred >= 0.5).astype(int)

return {
    "accuracy": float(round(accuracy_score(y, y_pred_label), 4)),
    "precision": float(round(precision_score(y, y_pred_label), 4)),
    "recall": float(round(recall_score(y, y_pred_label), 4)),
    "f1": float(round(f1_score(y, y_pred_label), 4)),
    "roc_auc": float(round(roc_auc_score(y, y_pred), 4))
}
```

ДОДАТОК Г

Апробація результатів роботи

Research on Machine Learning Methods for Predicting Diabetes Development

Oleh V. Liapota[a] and Oleksii S. Nazarov[a]

[a] *Kharkiv National University of Radio Electronics, Nauky Ave. 14, Kharkiv, 61166, Ukraine*

Abstract

The rapid increase in the prevalence of diabetes worldwide necessitates the development of accurate and timely prediction tools to support early intervention and treatment. This paper presents a study of various machine learning methods applied to medical data to predict the risk of developing diabetes.

Keywords

Prediction, diabetes, machine learning, XGBoost, MLP, CNN, Logistic Regression, Decision Tree, Random Forest.

1. Introduction

Diabetes mellitus remains one of the most common chronic diseases globally, significantly affecting patients' quality of life and burdening healthcare systems. According to the World Health Organization, the number of people with diabetes is steadily increasing, emphasizing the need for early risk detection and timely medical intervention. Traditional diagnostic methods often rely on manual analysis of medical indicators, which is labor-intensive and prone to errors.

In recent years, machine learning (ML) methods have shown high efficiency in medical data analysis. They help detect hidden patterns and improve diagnostic accuracy. ML algorithms can learn from large multidimensional datasets and predict the likelihood of disease based on key risk factors — such as glucose level, body mass index (BMI), age, genetic predisposition, etc.

The goal of this project is to study and compare different machine learning algorithms for predicting the risk of developing diabetes. A modular software system was developed that integrates data preprocessing, model training, and prediction into a single platform convenient for medical professionals. Particular attention is paid to ensemble models, neural networks, and methods for handling imbalanced data, especially SMOTE.

2. Data Preprocessing

The effectiveness of machine learning models largely depends on the quality of data preparation. For the diabetes prediction task, several key data preprocessing steps were implemented: cleaning, normalization, sample balancing, and feature engineering.

2.1. Normalization

Medical indicators such as glucose level, BMI, and blood pressure have different value ranges, which can lead to some features dominating during model training. To mitigate this effect, min-max normalization was applied to scale all values to a common range of [0, 1], ensuring equal contribution from all features in model building.

Рисунок Г.1 – Вступ до публікації

2.2. Synthetic Minority Oversampling Technique

Most open medical datasets, including the Pima Indians Diabetes Database used in this study, are class-imbalanced — the number of non-diabetic cases significantly exceeds the number of diabetic cases. This imbalance hampers the model's ability to detect rare positive cases.

To address this, the SMOTE method [1] was used. It synthetically generates new minority class examples by interpolating between nearest neighbors in the multidimensional feature space. This creates a balanced dataset and improves the model's sensitivity to rare cases.

2.3. Feature Engineering

In addition to basic medical parameters, feature engineering was implemented — the process of creating new, more informative features from existing data. The synthesized features were chosen based on their proven importance in other studies [2]. The new features include:

- **glucose_BMI_ratio** — the ratio of glucose to BMI, potentially indicating abnormal conditions;
- **is_obese** — a binary indicator of obesity;
- **age_group** — categorical grouping of age to highlight risk categories;
- **HbA1c_level** — an estimated glycated hemoglobin level as a long-term glucose indicator.

This improved the dataset's informativeness and the generalization ability of the models.

3. Machine Learning Methods

To address the problem of diabetes risk prediction, several ML algorithms were applied, ranging from classical statistical approaches to modern neural network models. The selection was based on their suitability for tabular medical data, classification accuracy, and ability to handle complex multivariate relationships.

3.1. Logistic Regression

A baseline binary classification algorithm used to estimate the probability that an object belongs to one of two classes. In the context of diabetes prediction, it estimates the likelihood of a patient being at high risk.

Mathematical model:

$$P(y = 1|x|) = \frac{1}{1 + e^{-(w^T x + b)}} \quad (1)$$

where $x = [x_1, x_2, \ldots, x_n]$ — feature vector, w — weight vector, b — bias.

3.2. Decision Tree

A classification/regression model that splits input data into subgroups based on feature values. Each node represents a decision based on a feature threshold, and leaf nodes represent final classes (diabetic or not).

At each node, the feature and threshold are chosen to maximize purity. Entropy-based impurity measure:

Mathematical model:

$$Entropy(S) = -\sum_{i=1}^{c} p_i \log_2 \quad p_i \quad (2)$$

where S — data subset at a node, p_i — proportion of class i y S , c — number of classes.

Рисунок Г.2 – Огляд процесів підготовки даних та базових алгоритмів ML

Information gain from a split:

$$InformationGain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} \times Entropy(S) \quad (3)$$

where S — original dataset, A — feature to spli, S_v — subset after splitting by feature A.

3.3. Random Forest

An ensemble ML method that builds multiple decision trees on random data and feature subsets. Final predictions are aggregated through voting or averaging.

Mathematical model:

$$\dot{y} = mode\{h_1(x),\ h_1(x),\ \ldots,\ h_T(x)\} \quad (4)$$

where $h_i(x)$ — i-tree prediction.

3.4. XGBoost

An optimized gradient boosting algorithm that builds a sequence of weak learners, correcting errors from previous models. It incorporates regularization and speed optimizations.

Objective:

$$\zeta = \sum_{i=1}^{n} l(y_i,\ \dot{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \quad (4)$$

where l — logistic loss, $\Omega(f_k)$ — regularization term.

3.5. Multilayer Perceptron

A type of fully connected neural network composed of:

- Input layer;
- Hidden layers;
- Output layer.

Input vector:

$$\mathrm{x} = [x_1,\ x_2,\ \ldots,\ x_n] \in R^d \quad (5)$$

Input transformation:

$$h = \sigma\left(W^{(i)}x + b^{(1)}\right) \quad (6)$$

where $W^{(i)}$ — weights, $b^{(1)}$ — bias, σ — activation function.

4. Results

To compare the effectiveness of the ML methods, standard classification metrics were used: Accuracy, Precision, Recall, F1 Score, and ROC AUC. See Table 1.

Table 1
Metrics for ML Methods

Model	Accuracy	Precision	Recall	F1 Score	ROC AUC
Logistic Regression	84.81%	83.33%	87.50%	85.37%	91.54%

Рисунок Г.3 – огляд комплексним алгоритмів ML

Decision Tree	79.75%	74.00%	92.50%	82.22%	84.71%
Random Forest	86.08%	83.72%	90.00%	86.75%	94.78%
XGBoost	86.08%	82.22%	92.50%	87.06%	94.17%
MLP	82.28%	80.95%	85.00%	82.93%	90.96%
CNN	83.54%	81.40%	87.50%	84.34%	90.38%

All values are normalized within [00.00% – 100.00%]. Using a simple ranking method, the models' prospects were calculated:

- Logistic Regresion — 85.48;
- Decision Tree — 81.43;
- Random Forest — 86.91;
- XGBoost — 87.04;
- MLP — 83.14;
- CNN — 84.33;

The best-performing models were ensemble methods (Random Forest and XGBoost), which are well-suited for multicriteria decision-making and demonstrated high performance in medical tasks. Results are match and confirm similar research [3].

5. Acknowledgments

This work was prepared by Oleh Lyapota, Kharkiv National University of Radio Electronics, Ukraine.

6. References

[1] Alghamdi M, Al-Mallah M, Keteyian S, Brawner C, Ehrman J, et al. (2017) Predicting diabetes mellitus using SMOTE and ensemble machine learning approach: The Henry Ford ExercIse Testing (FIT) project. PLOS ONE 12(7): e0179805. https://doi.org/10.1371/journal.pone.0179805.

[2] Dagliati A, Marini S, Sacchi L, et al. Machine Learning Methods to Predict Diabetes Complications. Journal of Diabetes Science and Technology. 2018;12(2):295-302. doi:10.1177/1932296817706375.

[3] El-Sofany, Hosam, El-Seoud, Samir A., Karam, Omar H., Abd El-Latif, Yasser M., Taj-Eddin, Islam A. T. F., A Proposed Technique Using Machine Learning for the Prediction of Diabetes Disease through a Mobile App, International Journal of Intelligent Systems, 2024, 6688934, 13 pages, 2024. https://doi.org/10.1155/2024/6688934.

Рисунок Г.4 – Результати дослідження та використані джерела

ДОДАТОК Д

Експертний висновок результатів перевірки кваліфікаційної роботи на відповідність оформлення вимогам ДСТУ 3008: 2015

Експертний висновок результатів перевірки кваліфікаційної роботи

студент (посада) | програмної інженерії (кафедра) | ІПЗм-23-3 (група)

Ляпота Олег Владиславович
(прізвище, ім'я, по батькові)

Зауваження

Пункт ДСТУ 3008-2015	Зміст пункту	Сторінка кваліфікаційної роботи
1	2	3
	7.1 Загальні положення	
	7.3 Нумерація сторінок звіту	
	7.4 Нумерація розділів, підрозділів, пунктів, підпунктів	
	7.5 Рисунки	
	7.6 Таблиці	
	7.7 Переліки	
	7.8 Примітки	
	7.9 Виноски	
	7.10 Формули та рівняння	
	7.11 Посилання	
	7.13 Список авторів	
	7.14 Скорочення та умовні познаки	
	7.15 Додатки	

зауважень немає

Експерт _____ (підпис) Олена ОЛІЙНИК (прізвище, ініціали)

06.06.2025