

Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek

Računarstvo usluga i analiza podataka

SEMINARSKI RAD

„AI predviđanje prihoda“

Maja Lešnjaković

Patricija Peša

Osijek, 2025.

Sadržaj

1. Uvod	1
2. Opis problema	1
2.1. Korišteni podaci	1
2.2. Korišteni postupci strojnog učenja	1
3. Opis programskog rješenja	1
3.1. Model strojnog učenja	1
3.2. Način korištenja API-ja	1
3.3. Klijentska aplikacija	1
3.4. Dodatno	1
4. Zaključak	1
5. Poveznice i literatura	1

1. Uvod

Problemski zadatak koji se obrađuje u ovom projektu svodi se na predviđanje poslovnog rasta uzrokovanog implementacijom umjetne inteligencije (AI) u poslovanje. Motivacija proizlazi iz sve učestalije primjene AI tehnologija u različitim industrijama, poput marketinga i proizvodnje, što otvara pitanje: koliko se prihod zaista povećava kada tvrtka implementira umjetnu inteligenciju i koristi je za unapređenje poslovanja?

2. Opis problema

Primjena umjetne inteligencije u poslovanju donosi brojne prednosti, ali i izazove. Najčešće postavljano pitanje glasi: Koliko će se povećati prihodi ako uvedemo umjetnu inteligenciju u poslovanje? Predviđanje poslovnog rasta uz pomoć umjetne inteligencije predstavlja važan alat u strateškom planiranju.

Naravno, postoje slični sustavi u obliku:

- Alata za poslovnu analitiku (npr. IBM Watson),
- Prilagođenih enterprise rješenja,
- AI savjetnika za optimizaciju resursa.

Međutim, nije moguće izraditi sustav koji će s potpunom točnošću predvidjeti postotni rast temeljen na konkretnim metrikama usvajanja umjetne inteligencije, jer na taj rast utječu brojni drugi čimbenici – djelomično unutarnji, a djelomično vanjski – na koje tvrtka često nema izravan utjecaj.

2.1. Korišteni podaci

U sklopu ovog projekta korišten je skup podataka „Global AI Content Impact Dataset“ u formatu .csv, koji sadrži simulirane podatke o utjecaju primjene umjetne inteligencije u poslovnim kontekstima. Na slici ispod prikazano je nekoliko redaka iz dataset-a.

Year	Industry	AI Adoption ...	AI-Generate...	Job Loss Du...	Revenue Incr...	Human-AI C...	Top AI Tools ...	Regulation S...	Consumer Tr...	Market Shar...
2022	Media	44.29	33.09	16.77	46.12	74.79	Bard	Strict	40.77	18.73
2025	Legal	34.75	66.74	46.89	52.46	26.17	DALL-E	Strict	35.67	35.02
2022	Automotive	81.06	96.13	10.66	45.6	39.66	Stable Diffusion	Moderate	54.47	22.76
2021	Legal	85.24	93.76	27.7	78.24	29.45	Claude	Moderate	51.84	1.93
2021	Gaming	78.95	45.62	17.45	1.05	21.7	Midjourney	Strict	41.77	21.41
2021	Retail	66.95	47.72	0.86	27.58	64.42	ChatGPT	Lenient	68.14	8.09
2023	Media	68.23	6.14	6.2	53.13	53.16	Claude	Moderate	71.6	15.26
2023	Gaming	91.27	33.87	41.67	56.26	59.93	ChatGPT	Moderate	46.98	28.51
2025	Education	17.02	87.77	4.59	52.45	64.72	Claude	Moderate	60.55	22.7

Svaki zapis u skupu podataka obuhvaća sljedeće atribute:

- Postotak usvojenosti AI tehnologije (%) (AI Adoption Rate),
- Količina sadržaja generiranog umjetnom inteligencijom (TB/godišnje) (AI-Generated Content Volume),
- Postotak izgubljenih radnih mjesta zbog umjetne inteligencije (%) (Job Loss Due to AI),
- Stopa suradnje ljudi i umjetne inteligencije (%) (Human-AI Collaboration Rate),
- Povjerenje potrošača u umjetnu inteligenciju (%) (Consumer Trust in AI),
- Tržišni udio AI-orijentiranih tvrtki (%) (Market Share of AI Companies).

Ciljana varijabla u ovom istraživanju je predviđeni rast prihoda (%), koji predstavlja procjenu utjecaja usvajanja AI tehnologije na financijske pokazatelje poduzeća.

Podaci su učitani korištenjem biblioteke Pandas, a zatim su očišćeni, normalizirani i pripremljeni za treniranje modela unutar programskog okruženja Python.

2.2. Korišteni postupci strojnog učenja

Model je razvijen korištenjem Microsoft Azure Automatic ML platforme te unutar razvojnog okruženja Python, uz upotrebu biblioteke scikit-learn.

Tijekom razvoja modela testirano je više algoritama strojnog učenja kako bi se pronašlo najbolje rješenje. Cilj je bio minimizirati MAE (Mean Absolute Error), uz očuvanje stabilnosti i robusnosti predikcija. U tablici su prikazani usporedni rezultati nekoliko algoritama:

Algoritam	MAE (↓)	Prednosti	Nedostaci
Random Forest Regressor	2.8	Otporan na overfitting, ne traži skaliranje	Manje interpretabilan
Linear Regression	5.1	Brz, jednostavan, interpretabilan	Loš za nelinearne odnose
Decision Tree Regressor	3.6	Jednostavan, hvata nelinearnost	Lako se prenauči
Gradient Boosting Regressor	2.6	Vrlo precizan, stabilan	Spor, osjetljiv na šum
K-Nearest Neighbors	4.2	Jednostavan, neparametarski	Osjetljiv na outliere i dimenzije

Na temelju rezultata, iako Gradient Boosting daje nešto nižu pogrešku, Random Forest je odabran jer je brži za treniranje, manje osjetljiv na šum u podacima i lakši za integraciju u API. Također se bolje ponašao na neuređenim podacima i bio otporniji na prenaučenost.

Treniranje i evaluacija modela provedeni su korištenjem funkcije `train_test_split`, uz metrike poput MAE (Mean Absolute Error) – srednje apsolutne pogreške. Algoritam Random Forest odabran je zbog svoje visoke preciznosti i robusnosti, a kao jedan od najboljih za zadani problem prepoznat je i od strane Microsoft Azure Automatic ML sustava. Završni model spremljen je pomoću biblioteke `joblib` u `.pkl` formatu za kasniju primjenu.

Slike ispod prikazuju Job i ispis nekoliko modela iz Azure Automatic ML sustava nakon što je završio Job.

Algorithm name	Responsible AI	Normalized root mean squared error	Sampling	Created on	Duration	Hyperparameters
MaxAbsScaler, ExtremeRandomTrees		0.30200	100.00 %	Apr 10, 2025 5:00 PM	1m 5s	boo
MaxAbsScaler, ExtremeRandomTrees		0.30205	100.00 %	Apr 10, 2025 3:24 PM	56s	boo
StandardScalerWrapper, ExtremeRandomTrees		0.30254	100.00 %	Apr 10, 2025 4:41 PM	54s	boo
MaxAbsScaler, RandomForest		0.30272	100.00 %	Apr 10, 2025 3:15 PM	1m 4s	boo
StandardScalerWrapper, RandomForest		0.30278	100.00 %	Apr 10, 2025 5:39 PM	56s	boo
MaxAbsScaler, ExtremeRandomTrees		0.30288	100.00 %	Apr 10, 2025 5:02 PM	1m 3s	boo
MaxAbsScaler, RandomForest		0.30308	100.00 %	Apr 10, 2025 3:13 PM	1m 9s	boo

3. Opis programskog rješenja

Cjelokupno programsko rješenje sastoji se od sljedećih komponenti:

- Model strojnog učenja spremljen u .pkl formatu koristeći biblioteku joblib,
- Flask API, koji prima korisničke zahtjeve i vraća predviđene rezultate,
- Web sučelje za unos korisničkih podataka i prikaz predviđanja modela,
- Deploy-a rješenja putem platforme Render.com.

3.1. Model strojnog učenja

Model je treniran unutar okruženja Google Colab, budući da je Microsoft Azure u više navrata ograničavao izvođenje treniranja zbog resursnih ograničenja i vremenskih ograničenja vezanih uz besplatni plan. S druge strane, Google Colab pruža fleksibilnije okruženje za rad, jednostavnije upravljanje resursima te lakše otkrivanje i otklanjanje pogrešaka tijekom razvoja modela.

Model je treniran za regresijski zadatak – predviđanje postotnog rasta prihoda na temelju ulaznih značajki povezanih s primjenom umjetne inteligencije u poslovanju. Kao što je prethodno navedeno, za izgradnju modela korišten je algoritam Random Forest Regressor zbog svoje otpornosti na prenaučенost, visoke točnosti te relativno dobre interpretabilnosti u usporedbi s drugim algoritmima strojnog učenja. Random Forest kombinira rezultate većeg broja stabala odluke, čime se smanjuje varijanca modela i povećava pouzdanost predviđanja.

Model je evaluiran pomoću metrike MAE (Mean Absolute Error) te je odabran kao završno rješenje na temelju uravnoteženog omjera točnosti, stabilnosti i izvedbe u testnom skupu podataka.

Učitavanje potrebnih biblioteka:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error
import joblib
```

Učitavanje podataka i odabir značajki:

```
features = [
    'AI Adoption Rate (%)',
    'AI-Generated Content Volume (TBs per year)',
    'Job Loss Due to AI (%)',
    'Human-AI Collaboration Rate (%)',
    'Consumer Trust in AI (%)',
    'Market Share of AI Companies (%)'
]
target = 'Revenue Increase Due to AI (%)'

X = df[features]
y = df[target]
```

Podjela podataka na trening i test skup:

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
```

Treniranje modela:

```
model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
```

Evaluacija modela:

```
y_pred = model.predict(X_test)
mae = mean_absolute_error(y_test, y_pred)
print(f"MAE: {mae:.2f}")
```

Spremanje treniranog modela:

```
joblib.dump(model, "ai_revenue_model.pkl")
```

Slika ispod prikazuje izlaz u konzoli prilikom treniranja modela.

```
(venv) C:\Users\Vaja\Downloads\train_model>pip install -r requirements.txt
Collecting pandas (from -r requirements.txt (line 1))
  Downloading pandas-2.2.3-cp311-cp311-win_amd64.whl.metadata (19 kB)
Collecting scikit-learn==1.3.2 (from -r requirements.txt (line 2))
  Using cached scikit_learn-1.3.2-cp311-cp311-win_amd64.whl.metadata (11 kB)
Collecting joblib (from -r requirements.txt (line 3))
  Using cached joblib-1.4.2-py3-none-any.whl.metadata (5.4 kB)
Collecting numpy<2.0,>=1.17.3 (from scikit-learn==1.3.2->-r requirements.txt (line 2))
  Downloading numpy-1.26.4-cp311-cp311-win_amd64.whl.metadata (61 kB)
----- 61.0/61.0 kB 1.1 MB/s eta 0:00:00
Collecting scipy>=1.5.0 (from scikit-learn==1.3.2->-r requirements.txt (line 2))
  Using cached scipy-1.15.2-cp311-cp311-win_amd64.whl.metadata (60 kB)
Collecting threadpoolctl>=2.0.0 (from scikit-learn==1.3.2->-r requirements.txt (line 2))
  Using cached threadpoolctl-3.6.0-py3-none-any.whl.metadata (13 kB)
Collecting python-dateutil>=2.8.2 (from pandas->-r requirements.txt (line 1))
  Downloading python_dateutil-2.9.0.post0-py2.py3-none-any.whl.metadata (8.4 kB)
Collecting pytz>=2020.1 (from pandas->-r requirements.txt (line 1))
  Downloading pytz-2025.2-py2.py3-none-any.whl.metadata (22 kB)
Collecting tzdata>=2022.7 (from pandas->-r requirements.txt (line 1))
  Downloading tzdata-2025.2-py2.py3-none-any.whl.metadata (1.4 kB)
Collecting six>=1.5 (from python-dateutil>=2.8.2->pandas->-r requirements.txt (line 1))
  Downloading six-1.17.0-py2.py3-none-any.whl.metadata (1.7 kB)
Using cached scikit_learn-1.3.2-cp311-cp311-win_amd64.whl (0.2 MB)
Downloading pandas-2.2.3-cp311-cp311-win_amd64.whl (11.6 MB)
----- 11.6/11.6 MB 5.2 MB/s eta 0:00:00
Using cached joblib-1.4.2-py3-none-any.whl (301 kB)
Downloading numpy-1.26.4-cp311-cp311-win_amd64.whl (15.8 MB)
----- 15.8/15.8 MB 6.2 MB/s eta 0:00:00
Downloading python_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)
----- 229.9/229.9 kB 4.7 MB/s eta 0:00:00
Downloading pytz-2025.2-py2.py3-none-any.whl (509 kB)
----- 509.2/509.2 kB 5.3 MB/s eta 0:00:00
Using cached scipy-1.15.2-cp311-cp311-win_amd64.whl (41.2 MB)
Using cached threadpoolctl-3.6.0-py3-none-any.whl (18 kB)
Downloading tzdata-2025.2-py2.py3-none-any.whl (347 kB)
----- 347.8/347.8 kB 7.2 MB/s eta 0:00:00
Downloading six-1.17.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: pytz, tzdata, threadpoolctl, six, numpy, joblib, scipy, python-dateutil, scikit-learn, pandas
Successfully installed joblib-1.4.2 numpy-1.26.4 pandas-2.2.3 python-dateutil-2.9.0.post0 pytz-2025.2 scikit-learn-1.3.2 scipy-1.15.2 six-1.17.0 threadpoolctl-3.6.0 tzdata-2025.2

[notice] A new release of pip is available: 24.0 -> 25.0.1
[notice] To update, run: python.exe -m pip install --upgrade pip

(venv) C:\Users\Vaja\Downloads\train_model>python train.py
MAE: 19.44
Model saved as ai_revenue_model.pkl
```

Slika ispod prikazuje ispis u konzoli prilikom build-anja modela.

```
Apr 13 11:05:42 PM [i] Using cached threadpoolctl-3.6.0-py3-none-any.whl (18 kB)
Apr 13 11:05:42 PM [i] Using cached werkzeug-3.1.3-py3-none-any.whl (224 kB)
Apr 13 11:05:42 PM [i] Using cached packaging-24.2-py3-none-any.whl (65 kB)
Apr 13 11:05:42 PM [i] Using cached MarkupSafe-3.0.2-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (23 kB)
Apr 13 11:05:42 PM [i] Using cached six-1.17.0-py2.py3-none-any.whl (11 kB)
Apr 13 11:05:42 PM [i] Installing collected packages: pytz, threadpoolctl, six, packaging, numpy, MarkupSafe, joblib, itsdangerous, click, blinker, Werkzeug, scipy, python-dateutil, Jinja2, gunicorn, scikit-learn, pandas, Flask, flask-cors
Apr 13 11:05:55 PM [i] Successfully installed Flask-2.3.2 Jinja2-3.1.6 MarkupSafe-3.0.2 Werkzeug-3.1.3 blinker-1.9.0 click-8.1.8 flask-cors-4.0.0 gunicorn-21.2.0 itsdangerous-2.2.0 joblib-1.3.2 numpy-1.24.4 packaging-24.2 pandas-1.5.3 python-dateutil-2.9.0.post0 pytz-2025.2 scikit-learn-1.3.2 scipy-1.15.2 six-1.17.0 threadpoolctl-3.6.0
Apr 13 11:05:55 PM [i] [notice] A new release of pip is available: 24.0 -> 25.0.1
Apr 13 11:05:55 PM [i] [notice] To update, run: pip install --upgrade pip
Apr 13 11:06:00 PM [i] ==> Uploading build...
Apr 13 11:06:09 PM [i] ==> Uploaded in 6.1s. Compression took 2.4s
Apr 13 11:06:11 PM [i] ==> Build successful 🎉
Apr 13 11:06:14 PM [i] ==> Deploying...
```

3.2. Način korištenja API-ja

Aplikacijsko programsko sučelje (API) realizirano je pomoću Flask okvira u programskom jeziku Python. API omogućuje komunikaciju između korisničkog sučelja i modela strojnog učenja, te je osmišljen kako bi primao podatke, obrađivao ih kroz trenirani model i vraćao rezultat predviđanja.

API definira dvije osnovne rute:

- "/" – vraća jednostavnu poruku koja potvrđuje da je API aktivan i dostupan,
- "/predict" – prima JSON objekt koji sadrži šest značajki (ulaznih vrijednosti) te vraća predviđenu vrijednost rasta prihoda.

Učitavanje modela:

```
import joblib
model = joblib.load("ai_revenue_model.pkl")
```

Osnovna struktura Flask aplikacije:

```
from flask import Flask, request, jsonify
from flask_cors import CORS
```

```
app = Flask(__name__)
CORS(app)
```

Ruta koja potvrđuje da je API aktivan:

```
@app.route('/')
def home():
    return "AI Revenue Prediction API is running!"
```

Ruta koja prima podatke i vraća predviđanje:

```
@app.route('/predict', methods=['POST'])
def predict():
    data = request.get_json()
    features = data.get("features", [])
    prediction = model.predict([features])
    return jsonify({"prediction": prediction[0]})
```

Model i API su postavljeni na Render.com, što je omogućilo besplatno hostanje aplikacije. Korištena je konfiguracijska datoteka „render.yaml“ koja specificira način pokretanja:

```
services:
  - type: web
    name: ai-flask-api
    env: python
    buildCommand: pip install -r requirements.txt
    startCommand: gunicorn app:app
```

Pomoću CORS (Cross-Origin Resource Sharing) omogućeno je da HTML forma iz drugog izvora može komunicirati s API-jem. Kada korisnik unose podatke, forma ih šalje kao JSON, a odgovor modela odnosno predviđanje, se prikazuje u stvarnom vremenu.

Online API adresa: <https://ai-flask-api.onrender.com>

Slike ispod prikazuju ispis konzole kada je API pokrenut i kada je u running-u.

```
Apr 13 11:12:17 PM  ➡ Docs on specifying a port: https://render.com/docs/web-services#port-binding
Apr 13 11:14:40 PM  127.0.0.1 - - [13/Apr/2025:21:14:40 +0000] "OPTIONS /predict HTTP/1.1" 200 0 "-" Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/135.0.0.0 Safari/537.36 Edg/135.0.0.0
Apr 13 11:14:40 PM  127.0.0.1 - - [13/Apr/2025:21:14:40 +0000] "POST /predict HTTP/1.1" 200 34 "-" Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/135.0.0.0 Safari/537.36 Edg/135.0.0.0
Apr 13 11:30:40 PM  [2025-04-13 21:30:40 +0000] [76] [INFO] Handling signal: term
Apr 13 11:30:40 PM  [2025-04-13 21:30:40 +0000] [91] [INFO] Worker exiting (pid: 91)
Apr 13 11:30:42 PM  [2025-04-13 21:30:42 +0000] [76] [INFO] Shutting down: Master
Apr 13 11:30:42 PM  ✅ Flask API je pokrenut.
Apr 13 11:30:42 PM  📊 Model očekuje značajke u sljedećem redoslijedu:
Apr 13 11:30:42 PM  1. AI Adoption Rate (%)
Apr 13 11:30:42 PM  2. AI-Generated Content Volume (TBs per year)
Apr 13 11:30:42 PM  3. Job Loss Due to AI (%)
Apr 13 11:30:42 PM  4. Human-AI Collaboration Rate (%)
Apr 13 11:30:42 PM  5. Consumer Trust in AI (%)
Apr 13 11:30:42 PM  6. Market Share of AI Companies (%)
```



```
Apr 13 11:06:54 PM ➡ running gunicorn app:app
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [76] [INFO] Starting gunicorn 21.2.0
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [76] [INFO] Listening at: http://0.0.0.0:10000 (76)
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [76] [INFO] Using worker: sync
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [91] [INFO] Booting worker with pid: 91
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [91] [INFO] Flask API je pokrenut.
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [91] [INFO] Model očekuje značajke u sljedećem redoslijedu:
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [91] [INFO] 1. AI Adoption Rate (%)
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [91] [INFO] 2. AI-Generated Content Volume (TBs per year)
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [91] [INFO] 3. Job Loss Due to AI (%)
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [91] [INFO] 4. Human-AI Collaboration Rate (%)
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [91] [INFO] 5. Consumer Trust in AI (%)
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [91] [INFO] 6. Market Share of AI Companies (%)
Apr 13 11:07:12 PM [2025-04-13 21:07:12 +0000] [91] [INFO] 127.0.0.1 - - [13/Apr/2025:21:07:12 +0000] "HEAD / HTTP/1.1" 200 0 "-" "Go-http-client/1.1"
Apr 13 11:07:15 PM [2025-04-13 21:07:15 +0000] [91] [INFO] ➡ Your service is live 🎉
Apr 13 11:07:17 PM [2025-04-13 21:07:17 +0000] [91] [INFO] 127.0.0.1 - - [13/Apr/2025:21:07:17 +0000] "GET / HTTP/1.1" 200 37 "-" "Go-http-client/2.0"
```

📖 ⬅ ➡ ✕ 🔒 https://ai-flask-api.onrender.com

AI Revenue Prediction API is running!

3.3. Klijentska aplikacija

HTML forma koristi JavaScript za pozivanje API-ja:

```
fetch('https://ai-flask-api.onrender.com/predict', {
  method: 'POST',
  headers: { 'Content-Type': 'application/json' },
  body: JSON.stringify({ features: [70, 100, 10, 60, 80, 25] })
})
.then(response => response.json())
.then(data => console.log(data.prediction));
```

3.4. Dodatno

Izgled web stranice prije i nakon što su poslani podaci i primljeno je predviđanje od strane modela.

AI Revenue Prediction

AI Adoption Rate (%):

AI-Generated Content Volume (TBs per year):

Job Loss Due to AI (%):

Human-AI Collaboration Rate (%):

Consumer Trust in AI (%):

Market Share of AI Companies (%):

Send Prediction

The screenshot shows a web application titled "AI Revenue Prediction" with a robot icon. It features a central form with six input fields, each with a label and a value: "AI Adoption Rate (%)" with 65, "AI-Generated Content Volume (TBs per year)" with 158, "Job Loss Due to AI (%)" with 2, "Human-AI Collaboration Rate (%)" with 58, "Consumer Trust in AI (%)" with 65, and "Market Share of AI Companies (%)" with 13. Below the inputs is a blue "Send Prediction" button. At the bottom of the form, it displays "Predikcija: 33.40". The interface has a dark blue background with a grid pattern.

4. Zaključak

U ovom projektu razvijen je sustav za predviđanje poslovnog rasta temeljen na analizi utjecaja umjetne inteligencije u poslovnim procesima. Korišten je simulirani skup podataka koji obuhvaća ključne pokazatelje vezane uz usvajanje AI tehnologije, poput stope automatizacije, suradnje ljudi i AI sustava te povjerenja potrošača.

Model strojnog učenja treniran je u okruženju Google Colab korištenjem algoritma Random Forest Regressor, koji se pokazao najpouzdanijim u pogledu točnosti i robusnosti. Implementirano je web API sučelje temeljeno na Flasku, koje omogućuje unos podataka i dohvat predikcija u stvarnom vremenu, a cijelo je rješenje uspješno postavljeno na platformu Render.com.

Projekt potvrđuje potencijal primjene strojnog učenja u strateškom odlučivanju i ukazuje na važnost dostupnosti kvalitetnih podataka za dobivanje pouzdanih rezultata. U budućnosti bi se preciznost modela mogla dodatno poboljšati proširenjem skupa podataka stvarnim poslovnim podacima te uvođenjem naprednijih algoritama i metoda optimizacije.

5. Poveznice i literatura

<https://dashboard.render.com/web/srv-cvu1tvre5dus73cf4e1g/deploy/dep-cvu2duidbo4c739dc850>

<https://ai-flask-api.onrender.com/>

<https://www.kaggle.com/>

<https://github.com/mlesnjakovic/ai-flask-api/tree/main>