

Dokumentace k projektu

Základní informace

Název projektu: Předpověď cen automobilů pomocí regrese

Autor: Filip Novotný

Kontaktní údaje: filipnovotny0902@gmail.com

Datum vypracování: 5. dubna 2025

Název školy: Střední průmyslová škola elektrotechnická, Praha 2, Ječná 30

Tento dokument byl vytvořen v rámci školního projektu.

Specifikace požadavků uživatele / Use Case

Tato část dokumentace popisuje, jaké údaje uživatel zadává do aplikace a jaké funkce aplikace vykonává.

Uživatel interaguje s aplikací prostřednictvím rozbalovacích (dropdown) menu a textových polí. Konkrétně uživatel:

- **Vybere značku:** Z dropdown menu, kde jsou uvedeny dostupné značky vozidel.
- **Vybere model:** Z dynamicky generovaného seznamu modelů, který se mění podle zvolené značky.
- **Vybere objem (l):** Z dropdown menu obsahujícího různé možnosti objemu motoru.
- **Zadá najeté kilometry:** Do textového pole.
- **Vybere palivo:** Z dropdown menu s nabídkou typů paliva.
- **Vybere převodovku:** Z dropdown menu.
- **Zadá výkon (kW):** Do textového pole.
- **Zadá stáří vozidla:** Do textového pole.

Návrhové vzory v projektu

1. Oddělení zodpovědností (Separation of Concerns):

Aplikace je rozdělena do samostatných modulů, kde každý modul plní specifickou úlohu (např. web scraping, předzpracování dat, trénink modelu, uživatelské rozhraní). Tento přístup usnadňuje údržbu a umožňuje nezávislý vývoj jednotlivých částí.

2. Facade Pattern (Fasádový vzor):

Hlavní skript aplikace (`app.py`) funguje jako jednotné rozhraní, které skrývá složitost interních procesů (načítání modelů, zpracování vstupů, předpověď ceny) a poskytuje uživateli jednoduché a intuitivní rozhraní pro interakci s aplikací.

3. Strategy Pattern (Strategický vzor):

Aplikace umožňuje volbu mezi různými modely strojového učení (Decision Tree, Random Forest, Gradient Boosting), přičemž každý model představuje odlišnou strategii pro předpověď ceny. Tento vzor umožňuje dynamicky vybírat optimální metodu podle požadavků a výsledků.

Popis chodu aplikace pomocí UML behaviorálních diagramů

Tato část dokumentace popisuje dynamiku a chování aplikace pomocí UML behaviorálních diagramů, které ilustrují, jak aplikace reaguje na vstupy a jak probíhá její interní logika.



Konfigurace aplikace a dostupné konfigurační volby

Aplikace je konfigurována pomocí souboru `.env`, který umožňuje snadnou úpravu nastavení bez nutnosti měnit samotný kód. Níže jsou popsány hlavní konfigurační volby a jejich účel:

- **BRAND:**
Určuje značku vozidla, pro kterou se mají získávat data. Například "skoda", "audi", "bmw".
- **NUM_LISTINGS:**
Udává minimální počet inzerátů, které má aplikace načíst. Pomocí této volby lze zajistit, že bude získáno dostatečné množství dat pro analýzu a trénink modelu.
- **MAX_PAGES:**
Nastavuje maximální počet stránek, které se mají procházet při získávání dat.
- **MAX_WORKERS:**
Definuje počet paralelních vláken, které se využívají při web scrapingu.
- **OUTPUT_DIR:**
Určuje cestu, kam se ukládají výsledné soubory (CSV, modely atd.).
- **MIN_PRICE a MAX_PRICE:**
Slouží jako cenové filtry při získávání dat z webu.

Instalace a spuštění aplikace

1. **Stahování a rozbalení projektu:**
Stáhněte si archiv projektu a rozbalte jej do vámi zvolené složky.
2. **Instalace závislostí:**
Otevřete terminál v kořenovém adresáři projektu a nainstalujte potřebné knihovny příkazem:

```
pip install -r requirements.txt
```
3. **Konfigurace aplikace:**
Upravte soubor `.env` podle svých potřeb (např. BRAND, NUM_LISTINGS, MAX_PAGES, MAX_WORKERS, OUTPUT_DIR, MIN_PRICE, MAX_PRICE).
4. **Spuštění aplikace:**
Spustěte hlavní skript (např. `app.py`) ve složce **src**: `python src/app.py`

Aplikace se otevře a zobrazí grafické uživatelské rozhraní, kde můžete zadat vstupní údaje a získat předpověď ceny vozidla.

Chybové stavy a postup řešení

- **Neplatné vstupy:**

Aplikace zobrazí chybová hlášení jako `"Chyba: Zadejte platné číselné hodnoty"` nebo `"Chyba: Zadejte pouze nezáporné hodnoty."`

- **Chyba při načítání dat:**

Pokud aplikace nedokáže načíst CSV soubory nebo jiné zdroje dat, zobrazí se zpráva

`"Chyba při načítání datasetu: [detaily chyby]"`.

- **Chyby při načítání modelů:**

Pokud aplikace nedokáže načíst předtrénované modely, vyhodí se hláška

`"Chyba při načítání modelu [název souboru]: [detaily chyby]"`.

- **Interní chyby aplikace:**

Například nesprávné nastavení konfigurace. Aplikace vypíše podrobný popis chyby pro snadnou diagnostiku.

Výčet knihoven třetích stran

- **pandas** – Pro manipulaci s daty, načítání a ukládání CSV.
- **numpy** – Pro podporu práce s numerickými daty a maticemi.
- **scikit-learn** – Pro modelování (Decision Tree, Random Forest, Gradient Boosting) a metriky (MSE, R^2 , MAE).
- **requests** – Pro HTTP požadavky (web scraping).
- **beautifulsoup4** – Pro parsování HTML a extrakci informací.
- **python-dotenv** – Pro načítání konfiguračních proměnných z `.env`.
- **joblib** – Pro ukládání a načítání modelů (kompatibilní se scikit-learn).
- **tkinter** – Pro tvorbu desktopového grafického rozhraní (GUI).
- **unicodedata2** – Pro práci s textem a odstranění diakritiky (normalizace).

Pro instalaci závislostí stačí spustit příkaz:

```
pip install -r requirements.txt
```

Závěrečné resumé projektu

Projekt "Předpověď cen automobilů" využívá regresi k předpovědi ceny vozidel na základě reálných dat získaných web scrapingem, čištěním a slučováním dat. Aplikace nabízí grafické uživatelské rozhraní pro zadávání specifikací vozidla a zobrazuje předpovězenou cenu. Celkové řešení je robustní a modulární, což zajišťuje pevný základ pro další rozvoj a optimalizaci.

Zdroje a reference

- [W3Schools Pandas Tutorial](#)
Základní návody k používání knihovny Pandas (DataFrame, CSV soubory).
- [YouTube: Python Pandas Tutorial](#)
Praktické ukázky načítání dat, práce s DataFrame a základních analýz.
- [Scikit-learn Documentation](#)
Oficiální dokumentace scikit-learn poskytuje detailní informace o modelování a hodnocení regresních modelů.

Web scraping a analýza cílových stránek:

- [autoesa.cz](#) - Zdroj reálných dat o automobilech.
- [sauto.cz](#) - Další zdroj reálných dat o automobilech.

Použité nástroje a konzultace

Pro návrh a optimalizaci kódu jsem využil ChatGPT jako pomocníka, který mi pomohl lépe pochopit a vysvětlit jednotlivé části kódu.