

Badania podstawowe – TLR poziom I

Predykcja zasięgu EV z korekcją stylu jazdy

Zespół: Piotr Zienowicz, Konrad Zieliński
Kierownik projektu: Konrad Zieliński

15 października 2025

Spis treści

1	Lab 2 — TLR poziom I	2
2	Lab 3 — TLR poziom II	3

1 Lab 2 — TLR poziom I

Opis zadania (TLR-1)

Tutaj rodzi się koncepcja. Celem TLR-1 jest sformułowanie głównej idei, która ma potencjał, lecz nie ma jeszcze jasno określonego ostatecznego zastosowania czy produktu. Dopuszczalne jest wskazanie wstępnych, potencjalnych zastosowań, które mogą zostać rozwinięte na kolejnych poziomach TLR.

Główna idea

Proponujemy lekką metodę **predykcji zasięgu pojazdów elektrycznych (EV)**, która łączy *sygnały stylu jazdy kierowcy* (np. zmienność prędkości, intensywność hamowań/przyspieszeń) z *warunkami zewnętrznymi* (pogoda, nachylenie trasy). Hipoteza: wzbogacenie informacji wejściowej o te czynniki pozwoli ograniczyć przeszacowania i niedoszacowania zasięgu względem prostych metod bazowych opartych jedynie o średnie zużycie energii.

Metryki (przykład: MAE)

Do oceny jakości wstępnych predykcji stosujemy m.in. średni błąd bezwzględny **MAE** (Mean Absolute Error):

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|, \quad (1)$$

gdzie y_i to wartość rzeczywista (np. zasięg w km), a \hat{y}_i – wartość przewidywana. MAE wyrażony jest w tej samej jednostce co mierzona wielkość (np. km).

Przykład Rzeczywiste zasięgi: [200, 180, 220], przewidywane: [190, 175, 210] \Rightarrow błędy bezwzględne: [10, 5, 10], a zatem $\text{MAE} = 8,33$ km.

Narracja TLR-1 (styl opisu)

W wyniku prac badawczych oraz analiz informatycznych, zespół [TBD] opracował lekką metodę predykcji zasięgu EV, łączącą sygnały stylu jazdy z warunkami zewnętrznymi. Zaobserwowano, że dokładność prognoz (MAE) jest lepsza o ok. 12–18% względem podejścia bazowego opartego wyłącznie o średnie zużycie energii z ostatnich kilometrów. Ponadto metoda wykazuje adaptację do zmiennych warunków (wiatr, temperatura) i różnic w topografii, co przekłada się na mniejsze przeszacowania i niedoszacowania. W kolejnych etapach (TLR-2) planowana jest szersza walidacja na odmiennych modelach EV i w różnych warunkach.

2 Lab 3 — TLR poziom II

Projekt: Predykcja zasięgu EV z korekcją stylu jazdy

Zadanie 1. Koncepcja technologii i jej przyszłe zastosowania

Narracja Przeprowadzono wstępną analizę możliwych zastosowań technologii predykcji zasięgu EV. Ustalono, że rozwiązanie powinno w **pierwszej kolejności** zostać wdrożone w **systemach pokładowych pojazdów (IVI/OS OEM)**, aby predykcja działała lokalnie i bez opóźnień. Zewnętrzne aplikacje mobilne (Android/iOS) oraz panel webowy dla flot mogą zostać przygotowane jako **kolejne etapy rozwoju**, po ugruntowaniu komponentu pokładowego. Optymalnym rozwiązaniem jest prosty, intuicyjny interfejs w pojeździe, minimalizujący liczbę kroków do uzyskania prognozy.

Zauważono, że komponent predykcyjny po odpowiedniej adaptacji może zostać rozszerzony o moduł *eco-suggestions* (mikrokorekty stylu jazdy) oraz moduł planowania ładowań (rekomendacja postojów i mocy ładowarek). Moduł predykcji zasięgu stanowi kluczowy komponent aplikacji produkcyjnej (najpierw pokładowej), natomiast pozostałe moduły mogą być włączane w kolejnych iteracjach lub w aplikacjach zewnętrznych.

Priorytety wdrożeniowe

1. Wdrożenie i optymalizacja komponentu predykcji w systemie pokładowym (IVI/OS OEM) – inference lokalnie, niski narzut obliczeniowy.
2. Integracja z czujnikami/telemetrią pojazdu (SoC, prędkość, ewentualnie OBD) oraz danymi o nachyleniu i pogodzie (cache/offline, gdy brak łączności).
3. Etap 2: lekkie aplikacje zewnętrzne (Android/iOS) do zbierania danych pilotażowych i wizualizacji; panel webowy dla flot.

Zakres technologiczny (kierunek rozwoju)

- Platformy docelowe: **IVI/OS OEM (priorytet)**, następnie Android/iOS oraz web (dashboard floty).
- Architektura: komponent pokładowy (zbieranie danych, inference, prezentacja alertów); usługa modelowania (API) do aktualizacji modelu i wymiany konfiguracji; panel webowy do metryk i historii (etap późniejszy).
- Integracje: API pogodowe (Open-Meteo, z cache); źródła wysokości (SRTM/OSM); opcjonalnie OBD (SoC/telemetria); mapy/trasowanie (OSM/komercyjne).
- UX: w pojeździe minimalna liczba kroków, przewidywalna latencja, tryb offline.

Przyszłe zastosowania

- Kierowca indywidualny: planowanie trasy i redukcja *range anxiety*.
- Floty/logistyka: planowanie kursów i ładowań, szacunek kosztów energii.
- Eco-driving: rekomendacje mikrokorekt stylu dla zmniejszenia zużycia.

Zadanie 2. Lista (podstawowa) rzeczy potrzebnych do dalszego rozwoju projektu

Infrastruktura i sprzęt

- Stabilny dostęp do Internetu (preferowane łącze światłowodowe).
- 2 komputery do programowania.
- 2 smartfony (Android/iOS) do testów i zbierania danych.
- 1–2 interfejsy OBD–II (dongle) – opcjonalnie, do pobierania SoC/telemetrii.
- Repozytorium/pamięć masowa na dane.

Oprogramowanie i usługi

- Środowisko: Python (pandas, scikit–learn, lightgbm/xgboost), Jupyter.
- System kontroli wersji: Git + repo.
- Issue tracker i komunikacja: Jira, Slack/Teams.
- Hosting demo (lub serwer wewnętrzny).
- Dostęp do API: pogoda (Open–Meteo), mapy/elewacja (OSM/SRTM), ewentualnie trasowanie.

Dane i testy

- Zbiór pilotażowy przejazdów (≥ 70 –100), różne profile tras i warunki pogodowe.
- Scenariusze testowe: miasto/trasa/mieszany; różne temperatury/wiatr; różne modele EV (jeśli możliwe).