

УДК 004.42

РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ ПРОГНОЗУВАННЯ ЧАСУ ДОСТАВКИ З ВИКОРИСТАННЯМ МЕТОДІВ МАШИННОГО НАВЧАННЯ

Мудрик Б.В., студент

Київський національний університет технологій та дизайну

Мельник Г.В., кандидат технічних наук, доцент

Київський національний університет технологій та дизайну

Ключові слова: машинне навчання, прогнозування часу доставки, градієнтний бустинг, LightGBM, ML.NET, геоінформаційні сервіси, логістика.

Сучасний ринок електронної комерції та доставки їжі потребує високої точності прогнозування часу виконання замовлень. Стандартні навігаційні сервіси (наприклад, Google Maps) обчислюють лише ідеальний транзитний час, ігноруючи специфічні бізнес-процеси: час приготування, досвід кур'єра та етап «останньої милі». З іншого боку, закритість алгоритмів великих агрегаторів та їх високі комісії обмежують можливості малого та середнього бізнесу. Тому розробка незалежного програмного модуля, що поєднує відкриті геодані з кастомними моделями машинного навчання, є надзвичайно актуальною.

Метою роботи є дослідження методів предиктивної аналітики та розробка інтелектуальної системи для точного розрахунку тривалості логістичних маршрутів.

Для розв'язання задачі регресії було обрано ансамблеві методи на базі дерев рішень, зокрема алгоритм градієнтного бустингу LightGBM[1]. На відміну від лінійних моделей, він автоматично виявляє складні нелінійні зв'язки (наприклад, синергію погодних умов та заторів), а на відміну від нейронних мереж - забезпечує високу швидкість навчання на табличних даних та інтерпретованість результатів.

Математична модель системи базується на комплексному рівнянні, де загальний час T є функцією від базового часу маршруту, часу приготування та суми додаткових затримок:

$$T = t_{base} + t_{prep} + \sum_{i=1}^n \Delta t_i$$

У роботі реалізовано низку інженерних рішень для підвищення надійності:

- Алгоритмічний фолбек: автоматична адаптація пішохідних маршрутів для велосипедного транспорту за допомогою

коефіцієнта швидкості $k=3.2$ у випадках відсутності велоінфраструктури.

- Концепція Just-in-Time: паралелізація процесів приготування страви та доїзду кур'єра до закладу, що дозволило уникнути систематичного завищення прогнозів.
- Механізм Guardrail: детермінований обмежувальний фільтр, що порівнює прогноз ML-моделі з абсолютним фізичним мінімумом часу, запобігаючи видачі нелогічних результатів.

Програмна реалізація здійснена на платформі .NET (мова C#) [3]. Використання фреймворку ML.NET дозволило інтегрувати предиктивне ядро безпосередньо в основний застосунок, уникаючи затримок, характерних для Python-мікросервісів [2]. Для взаємодії з геоданими використано Google Directions API [4], а для демонстрації результатів — графічний інтерфейс на базі WPF [5].

Результати тестування підтвердили ефективність обраного підходу:

- Навчена модель досягла коефіцієнта детермінації $R^2 = 0.9712$.
- Це свідчить про здатність системи пояснити понад 97% дисперсії цільової змінної.
- Час тренування моделі склав лише 3.73 секунди, що доводить високу обчислювальну ефективність архітектури.

Розроблений модуль є готовим інструментом для автоматизації логістики, що дозволяє бізнесу отримати точні прогнози без фінансової залежності від монополістів ринку.

Список використаних джерел

1. Ke G. LightGBM: A Highly Efficient Gradient Boosting Decision Tree / G. Ke, Q. Meng, T. Finley et al. // Advances in Neural Information Processing Systems. 2017. Vol. 30. P. 3146–3154.
2. ML.NET Guide: Machine Learning for .NET developers/ Microsoft Learn. 2024. URL:<https://learn.microsoft.com/en-us/dotnet/machine-learning/> (дата звернення: 16.04.2026).
3. Troelsen A. Pro C# 10 with .NET 6: Foundational Principles and Practices in Industrial Software Development / A. Troelsen, P. Japikse. 11th ed. Birmingham : Apress, 2022. 1184 p.
4. Google Maps Platform. Directions API Overview: Documentation. 2025. URL:<https://developers.google.com/maps/documentation/directions/overview> (дата звернення: 16.04.2026).
5. Windows Presentation Foundation (WPF) Desktop Guide [Електронний ресурс] / Microsoft Learn. 2024. URL: <https://learn.microsoft.com/en-us/dotnet/desktop/wpf/> (дата звернення: 16.04.2026).