Тема 19: Система анализа задач службы технической поддержки с авторекомендациями.

Чекпоинт 5 Нелинейные ML-модели

Команда 34:

- Ермаков Павел
- Ефимова Елена
- Рудаш Кирилл
- Филоненко Феодосия
- Ильин Илья

Куратор:

Каюмов Руслан

TS

1. Создание временных признаков: первый шаг включает в себя создание признаков на основе времени из столбца с датой. Это включает в себя извлечение атрибутов, таких как день недели, месяц, квартал и год. Эти признаки отражают временные закономерности и сезонность в данных.

2. Создание лаговых признаков: создаются лаговые признаки целевой переменной (количество новых заявок). Это включает в себя сдвиг целевой переменной на определенное количество временных периодов для создания новых признаков, представляющих прошлые значения. Этот шаг фиксирует временные зависимости и автокорреляцию во временных рядах. Пропущенные значения, возникающие в результате создания лагов, удаляются.

3. Обнаружение аномалий: обнаружение аномалий выполняется с использованием алгоритма Isolation Forest. Этот шаг выявляет точки данных, которые значительно отклоняются от нормальных закономерностей в каждой очереди. Алгоритм обучается на данных для выявления аномалий, и создается бинарный признак, указывающий, является ли точка данных аномалией или нет.

4. Создание признаков скользящей статистики: вычисляется скользящая статистика для целевой переменной. Это включает в себя вычисление статистических показателей, таких как среднее, медиана и стандартное отклонение, по скользящему окну указанного размера. Эти признаки отражают краткосрочные, среднесрочные тенденции и волатильность во временных рядах.

TS

В рамках пятого чекпоинта было проведено исследование градиентных бустингов для прогнозирования количества тикетов в очередях верхнего уровня технической поддержки.

- Сравнение трех алгоритмов градиентного бустинга:
  - CatBoost
  - LightGBM
  - XGBoost
- Тюнинг гиперпараметров с помощью Optuna:
  - Для каждой модели подбирались оптимальные значения гиперпараметров с использованием библиотеки Optuna.
- Кластеризация очередей верхнего уровня:
  - С использованием PCA для снижения размерности.
  - С помощью алгоритма K-Means для выделения групп очередей с похожим поведением.

NLP

В рамках пятого чекпоинта было проведено исследование градиентных бустингов для классификации тикетов.
Данные предварительно очищены, обработаны и токенизированы моделью Word2Vec.

Добавлены новые методы с возможностью изменения гиперпараметров для бустингов: Catboost, XGBoost, LightGBM, доработан Streamlit интерфейс, предоставляющий возможность обучения моделей с разными параметрами..
В ходе исследования в юпитер ноутбуках были подобраны наилучшие гиперпараметры для каждого из бустингов с помощью Optuna.

TS

Модель	Тип параметров	RMSE	MAE	MAPE	Время обучения
CatBoost	Default	194.2	88.3	0.57	33 сек
	Optuna	151.9	85.6	1.28	65 минут
XGBoost	Default	199.9	89.2	0.37	2 сек
	Optuna	144.3	74.6	0.58	10 сек
LightGBM	Default	164.8	86.1	0.45	1 сек
	Optuna	137.2	68.3	0.35	5 сек

NLP

1. ***Сравнение моделей по метрике ROC AUC***

Logistic Regression: 0.908

SVC: 0.935

CatBoost: 0.984; Optuna + CatBoost: 0.982

XGBoost: 0.971; Optuna + XGBoost: 0.968

Lightgbm: 0.981; Optuna + Lightgbm: 0.982

2. ***Сравнение по метрике accuracy***

Logistic Regression: 0.70

SVC: 0.75

CatBoost: 0.88; Optuna + CatBoost: 0.88

XGBoost: 0.86; Optuna + XGBoost: 0.86

Lightgbm: 0.87; Optuna + Lightgbm: 0.87

TS

Модель (кластеризация)	Тип параметров	RMSE	MAE	MAPE	Время обучения
LightGBM кластер 0	Default	105.8	48.8	0.44	1 сек
	Optuna	64.5	37.3	0.33	1 сек
LightGBM кластер 1	Default	302.8	176.1	0.26	1 сек
	Optuna	233.3	140	0.21	1 сек

TS

Изначально были обучены 3 модели бустинга (без кластеров), в итоге был выбран LightGBM, он показал наилучшие результаты после оптимизации Optuna.

Второй подход: разделение очередей верхнего уровня на 2 кластера (K=2) и обучение отдельных моделей LightGBM для каждого кластера. Однако, этот подход не улучшил результаты по сравнению с глобальной моделью.

В итоге была выбрана глобальная модель LightGBM, обученная на всех данных без учета кластеров.

NLP

Логистическая регрессия — худший вариант, плохо обрабатывает редкие классы. SVC показал улучшение, но все еще плохо работает с некоторыми редкими классами.

CatBoost однозначно лучшая модель, обеспечивающая наивысшую точность и сбалансированность по всем классам. Другие бустинги также показали отличный результат, незначительно ниже, чем Catboost. Оптимизация CatBoost через Optuna не дала явных улучшений, но подтвердила, что градиентный бустинг уже является оптимальным выбором.