**Инструкция пользователя**

**FEDOT.Industrial** — open-source фреймворк автоматического машинного обучения (AutoML), предназначенный для решения промышленных задач. Области применения:

1. Классификация (для временных рядов или изображений) (рисунок 1).



Рисунок 1 – Генераторы признаков

1. Обнаружение аномалий (для временных рядов или изображений).
2. Выявление переломных точек (для временных рядов).
3. Обнаружение объектов на изображениях.

На рисунке 2 приведена общая структура фреймворка. Она включает в себя блоки, предназначенные для решения частных задач создания композитных моделей и решения задач классификации, прогнозирования и выявления аномалий на данных случайных процессов.

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 2 – Структура фреймворка, реализующего алгоритмы для задач классификации, прогнозирования и выявления аномалий на данных случайных процессов

Задача классификации временных рядов решается с помощью блока генерации признаков различной природы (спектральных, вейвлетных, квантильных, топологических). Реализована также и возможность добавления пользовательских классов, реализующих специфичные для конкретной предметной области признаков. Далее используется модуль создания композитной предсказательной модели-классификатора (использующий для решения задачи фреймворк FEDOT) на извлеченных ранее признаков. Он основан на использовании оптимизатора графовых моделей GOLEM. В результате работы данных моделей компонент формирует пайплайн моделирования, решающий задачу классификации временных рядов.

Для задачи автоматического поиска аномалий в данных случайных процессов применяется блок оценки аномальности. Задача автоматического предсказательного моделирования на данных пространственно-временных полей решается с помощью дополнительных блоков аугментации и работы с предобученными нейросетевыми моделями.

Блок интерпретации модели и результатов моделирования позволяет визуализировать связь между использованными признаками и результат предсказания. Для обеспечения воспроизводимости экспериментов реализовано сохранение структуры пайплайна, весов входящих в его состав моделей и операций, а также самих результатов моделирования.

**Пример работы с фреймворком для решения задачи классификации временных рядов**

* **Шаг 1**. Инициализация экземпляра класса FedotIndustrial, и передача ему ряд именованных аргументов

from core.api.main import FedotIndustrial

industrial = FedotIndustrial(task='ts\_classification',

dataset='ItalyPowerDemand,

strategy='statistical',

use\_cache=True,

timeout=15,

n\_jobs=4,

window\_sizes='auto',

logging\_level=20,

output\_folder=None)

* **Шаг 2**. Загрузка данных и запуск эксперимента

train\_data, test\_data, \_ = industrial.reader.read(dataset\_name='ItalyPowerDemand')

model = industrial.fit(train\_features=train\_data[0], train\_target=train\_data[1])

labels = industrial.predict(test\_features=test\_data[0])

metric = industrial.get\_metrics(target=test\_data[1], metric\_names=['f1', 'roc\_auc'])

Параметры конфигурации:

* task – тип решаемой задачи (ts\_classification)
* dataset – имя набора данных для эксперимента
* strategy – способ решения задачи: конкретный генератор или в режиме fedot\_preset
* use\_cache – флаг для использования кеширования извлечённых признаков
* timeout – максимальное количество времени для составления пайплайна для классификации
* n\_jobs – количество процессов для параллельного выполнения
* window\_sizes – размеры окон для оконных генераторов
* logging\_level – уровень логирования
* output\_folder – путь к папке для сохранения результатов

Наборы данных должны храниться в каталоге data и разделяться на train и test с расширением .tsv. Имя папки в каталоге data должно соответствовать названию набора данных, который будет использоваться в эксперименте. В случае, если в локальной папке нет данных, класс Data Loader попытается загрузить данные из [архива UCR](https://www.cs.ucr.edu/~eamonn/time_series_data/).

Генераторы признаков, которые могут быть указаны в конфигурации:

quantile, wavelet, recurrence и topological.

Можно объединить несколько генераторов через присвоение полю strategy:

'ensemble: topological wavelet quantile'

**Компонент размещен в зеркалированных репозиториях:**

* <https://github.com/aimclub/Fedot.Industrial>
* <https://gitlab.actcognitive.org/itmo-nss-team/FEDOT-Industrial>