

Эволюционное автоматизированное машинное обучение для упрощенных мультимодальных пайплайнов

Гетманов Андрей, Никитин Николай
Университет ИТМО, г. Санкт-Петербург

Аннотация — Эффективное моделирование мультимодальных данных по-прежнему остается одной из сложных проблем современного машинного обучения. Большинство существующих инструментов для автоматического машинного обучения не учитывают мультимодальность, а некоторые другие используют вычислительно дорогие сквозные глубокие модели, которые могут быть неэффективными для многих случаев. В этой статье мы предлагаем эволюционный подход для автоматизированного проектирования легких мультимодальных пайплайнов. Эксперименты с мультимодальным тестом AutoML подтверждают, что предлагаемый подход позволяет достичь конкурентных показателей по сравнению с более сложными современными решениями.

Ключевые термины — Автоматическое машинное обучение, мультимодальность, эволюционная оптимизация

1. Введение

Автоматизированное машинное обучение (AutoML) стало популярным инструментом в научных и промышленных кругах благодаря своей способности упрощать разработку моделей машинного обучения. Несмотря на его широкое использование, возникает проблема при работе с наборами данных, которые содержат не только табличные элементы, но и неструктурированные данные, такие как текст и изображения. Одновременное использование многих источников данных (мультимодальные данные) может повысить качество решений сложных задач. Однако интеграция этих методов часто приводит к увеличению вычислительных затрат, особенно при оптимизации пайплайнов моделирования [1]. Эта задача является довольно ресурсоемкой, поэтому для получения результатов может потребоваться много времени. Это очевидное существенное узкое место в современных платформах AutoML, которое создает препятствия для беспрепятственной интеграции поддержки мультимодальных данных.

В современных автоматизированных системах для управления мультимодальными данными используются различные методы. Некоторые используют базовые методы, такие как векторизация n-gram или TF-IDF, для преобразования текстовых объектов в табличный формат [2], в то время как другие используют сложные нейронные сети для объединения различных типов данных в одно пространство объектов [3]. Выбор метода зависит от доступных ресурсов и используемых алгоритмов AutoML. В решениях AutoML используются различные способы поиска наилучшей модели, такие как случайный поиск, байесовская оптимизация и генетические алгоритмы [4]. Они также отличаются тем, как строятся модели: от простых последовательностей до более сложных комбинаций, таких как укладка и усиление [5], иногда образуя сложную структуру, известную как направленный ациклический граф [6].

Несмотря на эти достижения, в данной области отсутствует легкий и универсально применимый алгоритм для автоматизированного, открытого поиска пайплайнов в области композитного искусственного интеллекта. В данной работе предлагается новый и эффективный подход AutoML для построения моделей на мультимодальных наборах данных. Наш подход обеспечивает ряд преимуществ: последовательное обучение модели, кэширование промежуточных конфигураций пайплайна, параллельное оценивание подходящих функций и векторизация признаков с помощью нейронных сетей. Значимость предложенного метода заключается в том, что он способен существенно повысить вычислительную эффективность для широкого спектра приложений AutoML, основанных на эволюционных алгоритмах для таких задач, как регрессия или классификация.

Для эмпирической проверки эффективности нашего подхода мы провели серию экспериментов с использованием различных общедоступных наборов данных. Результаты, подробно описанные в разделе V, подтверждают нашу гипотезу: предложенный метод демонстрирует более высокие показатели качества по сравнению с существующими методами AutoML. Реализация этой методологии внесена в открытый фреймворк AutoML FEDOT, поэтому доступ к ней может получить любой представитель исследовательского сообщества.

Работа организована следующим образом: В разделе II описаны стратегии обработки мультимодальных данных, используемые в различных инструментах AutoML (включая SOTA). В разделе III представлена постановка задачи по улучшению алгоритма мультимодального обучения AutoML. В разделе IV предлагается набор новых улучшений для мультимодального эволюционного AutoML. В разделе V приведена экспериментальная оценка предложенных методов для различных примеров. Наконец, в разделе VI представлен анализ полученных результатов и возможное расширение исследования.

II. СМЕЖНЫЕ РАБОТЫ

Существует несколько решений AutoML, которые могут использовать не только табличные данные, но и работать с различными модальностями, такими как текст и изображения (иногда одновременно): AutoGluon [3], AutoSklearn [2], AutoKeras [7], LightAutoML [8], H2O [9], TPOT [10], HyperNets [11] и другие [12].

Как правило, они предлагают ограниченную поддержку для обработки изображений или текстов, часто используя несколько предварительно обученных моделей или требуя ручной предварительной обработки данных для извлечения признаков с помощью таких методов, как bag-of-words [13]. Распространенным мультимодальным случаем является сочетание табличных данных и текстовых полей [14]. Насколько нам известно, только AutoGluon поддерживает гибкую мультимодальность и работает с таблицами, текстами и изображениями как единое целое. Также доступна мультимодальная версия AutoGluon, основанная на фундаментальных моделях [15].

С другой стороны, известно, что промежуточное представление, основанное на комбинации данных из разных модальностей, может выявить более сложные зависимости данных и в результате дать более высокое качество решения [16], [17]. Это можно сформулировать как задачу обучения представлений [18].

Решения AutoML, основанные на эволюционных алгоритмах, могут опередить другие решения, основанные на подходах boosting и stacking, за счет нахождения нетривиальной структуры комбинаций моделей [11], [19], но эволюционный подход, как правило, более затратен в вычислениях, поэтому добавление большего числа модальностей может сделать процесс обучения невыгодным с точки зрения времени обучения и вычислительной мощности.

Одним из перспективных направлений в AutoML является нейронный поиск архитектуры (NAS) [20]. Оно может быть применимо и к мультимодальной задаче, поскольку данные всех модальностей можно объединить в одно векторное пространство [21] и использовать мультимодальные нейронные сети. У этого подхода есть одна большая проблема: наборы NAS и мультимодальных данных по отдельности требуют больших вычислительных затрат, поэтому подход, объединяющий их, потребует еще больше ресурсов.

Итак, существуют некоторые подходы к работе с мультимодальными данными в основанных на стеке AutoML-фреймворках, таких как AutoGluon, но использование эволюционных алгоритмов может, возможно, превзойти их как способ получения лучшего баланса между качеством и сложностью моделей. Одной из основных задач при использовании этого подхода с мультимодальными данными является повышение вычислительной эффективности процесса обучения. По этой причине мы решили сформулировать постановку задачи, специфичную для эволюционного автоматизированного машинного обучения для мультимодальных наборов данных.

III. ПОСТАНОВКА ЗАДАЧИ

Мы хотим реализовать поддержку мультимодальных данных в AutoML на основе эволюционных алгоритмов. Предлагаемый алгоритм должен поддерживать данные из разных источников (числовые, категориальные и текстовые признаки) и быть эффективным как с точки зрения качества, так и с точки зрения вычислительной эффективности, поэтому он должен не только превосходить существующие решения AutoML на открытых бенчмарках, но и создавать облегченные ансамбли моделей, способные быстро обучаться и занимать мало памяти. Этот алгоритм должен быть реализован в существующем эволюционном AutoML с открытым исходным кодом фреймворке FEDOT.

Сформулируем эту проблему в терминах оптимизационной задачи. Входные данные можно обозначить как X , результаты моделирования – как Y . Следует учитывать, что мультимодальные данные состоят не только из числовых признаков, поэтому в случае числовых и текстовых признаков X можно определить следующим образом:

$$X = \mathbb{R}^n \times \mathcal{T}, \quad (1)$$

где \mathcal{T} - множество строк.

Ансамбль моделей можно рассматривать как направленный ациклический граф, поэтому задача является многоцелевой: найти структуру и гиперпараметры графа,

которые дают наибольшую метрику качества и наименьшую метрику сложности. Таким образом, формулировка задачи оптимизации может быть определена следующим образом [22]:

$$M_{opt}^G = \operatorname{argmin}_{M^G} F(A_i, \{H_{A_i}\}_k, E_j), \quad (2)$$

$$F(M^G) = (Q(M^G), P(M^G)), \quad (3)$$

где F - векторная целевая функция, Q - критерий качества прогноза, полученного с помощью модели M^G , P - критерий вычислительной производительности модели. Для удобства чтения аргументы X и Y пропущены.

IV. ПРЕДЛАГАЕМЫЙ СПОСОБ

В этом разделе описан предлагаемый подход для мультимодального эволюционного AutoML. Мы предлагаем базовую стратегию эволюционного проектирования пайплайна и модифицированный подход "разделяй и властвуй". Кроме того, мы реализовали дополнительную систему обработки для нейронного извлечения признаков из текста.

A. Основной подход

Основной подход заключается в поиске структуры и гиперпараметров всей модели сразу и представлен на рисунке 1. Теоретически это может привести к поиску лучшего решения за счет более широкого пространства поиска, но на практике такие модели обучаются гораздо дольше и потребляют больше вычислительных ресурсов, что часто приводит к переполнению памяти и прекращению обучения.

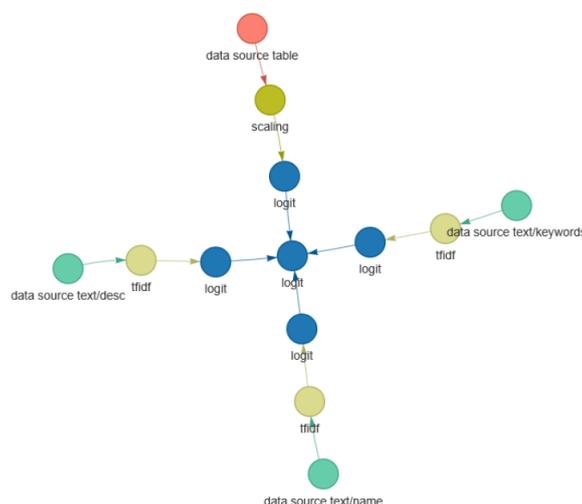


Рис. 1: Основной подход к разработке мультимодальных моделей

Подробное описание базового алгоритма приведено ниже. Алгоритм получает мультимодальные данные, автоматически идентифицирует текстовые колонки и разделяет данные на $n+1$ источников данных, где n - количество текстовых колонок (с дополнительной колонкой для источника табличных данных). На следующем шаге алгоритм создает начальный пайплайн, используя начальные предположения для каждого источника данных и узла ансамбля.

Исходным предположением для каждого текстового источника является TF-IDF плюс логистическая регрессия для задач классификации или TF-IDF плюс линейная регрессия для задач регрессии. Эти пайплайны обычно используются в качестве базовых во многих задачах благодаря оптимальному соотношению времени обучения и качества метрики. После определения пайплайнов для всех источников данных они собираются вместе с моделью логистической/линейной регрессии в качестве базовой, и алгоритм ищет оптимальную структуру пайплайна за оставшееся время. Это конечное время обозначается как тайм-аут в минутах. На последнем этапе вычисляется метрика качества, которую алгоритм выводит вместе с конечным пайплайном. Детальная реализация предложенного подхода описана в Алг. 1. Термин "граф" обозначает составной пайплайн, полученный в результате эволюционной оптимизации.

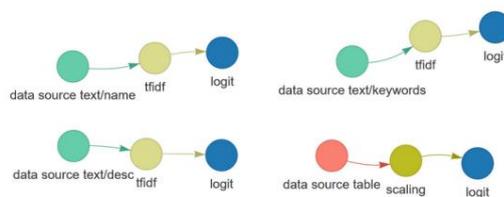
Algorithm 1 Basic Multimodal Training Algorithm

```
1: procedure TRAINMULTIMODALMODEL
2:   Input:
     data (dataset with text and table fields),
     task (regression or classification),
     target (target columns),
     initial_assumption (baseline graph),
     n (number of parallel jobs),
     timeout (overall time of search in minutes)
3:   Output: metric, pipeline
4:   data_sources  $\leftarrow$  DivideIntoDataSources(data, n)
5:   cache  $\leftarrow$  InitializeCache()
6:   ensembled_graph  $\leftarrow$  EnsembleModels(cache,
7:     task)
8:   final_graph  $\leftarrow$  FindBestFinalGraph(source, task,
9:     timeout, ensembled_graph)
10:  metric  $\leftarrow$  CalculateMetric(final_graph, data, target)
11:  return metric, final_graph
```

В. Усовершенствованный подход

Более сложная реализация предложенного подхода состоит из нескольких этапов. Сначала для каждого источника данных находится наилучшая структура и гиперпараметры модели, а затем аналогичным образом осуществляется поиск наилучшей конфигурации ансамблевой модели, которая принимает на вход предсказания ранее полученных моделей. Высокоуровневая схема подхода представлена на рисунке 2.

Step 1. Sequential training of submodels



Step 2. Ensemble model training

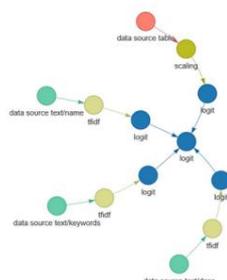


Рис. 2: Предлагаемый подход к улучшению вычислительной производительности для мультимодального AutoML

Алгоритм берет мультимодальные данные, автоматически идентифицирует текстовые колонки и разбивает данные на $n+1$ источников данных, где n - количество текстовых колонок (с дополнительной колонкой для источника табличных данных). На следующем этапе алгоритм обрабатывает каждый источник данных, чтобы определить оптимальную структуру пайплайна и гиперпараметры за отведенное время. Время, отведенное на обработку каждого источника, рассчитывается по уравнению 4.

$$T_{\text{source}} = \text{int} \left(\frac{\text{timeout}}{n + 1} \right) \quad (4)$$

Исходное предположение для каждого источника текста остается таким же, как и в базовом алгоритме. Впоследствии лучший пайплайн кэшируется. После определения пайплайнов для всех источников данных они объединяются с помощью логистической/линейной регрессии модель служит в качестве базовой, и алгоритм ищет оптимальную конечную структуру пайплайна за оставшееся время. Это конечное время обозначается как таймаут в минутах. На последнем этапе вычисляются метрики качества и сложности, и алгоритм выводит эти метрики вместе с конечным пайплайном. Подробная реализация предложенного подхода описана в Алг. 2.

Algorithm 2 Sequential Multimodal Training Algorithm

```
1: procedure TRAINMULTIMODALMODEL
2:   Input:
   data (dataset with text and table fields),
   task (regression or classification),
   target (target columns),
   initial_assumption (baseline graph),
   n (number of parallel jobs),
   timeout (overall time of search in minutes)
3:   Output: metric, pipeline
4:   data_sources  $\leftarrow$  DivideIntoDataSources(data, n)
5:   time_per_source  $\leftarrow$  int(timeout/(n + 1))
6:   cache  $\leftarrow$  InitializeCache()
7:   for each source in data_sources do
8:     best_graph  $\leftarrow$  FindBestGraph(source, task,
9:       time_per_source, initial_assumption)
10:    StoreInCache(cache, source, best_graph)
11:  ensembled_graph  $\leftarrow$  EnsembleModels(cache,
12:    task)
13:  final_graph  $\leftarrow$  FindBestFinalGraph(source, task,
14:    timeout, ensembled_graph)
15:  metric  $\leftarrow$  CalculateMetric(final_graph, data, target)
16:  return metric, final_graph
```

Предложенный подход реализован в рамках саморазвивающегося opensource-фреймворка FEDOT для эволюционного проектирования пайплайнов композитного моделирования.

Список возможных моделей (пространство поиска) для пайплайна (графа) состоит из обычных ML-алгоритмов для задач регрессии (Linear Regression, Ridge, Gradient Boosting Regressor и др.), классификации (Logistic Regression, Random Forest Classifier, KNN и др.) и операций преобразования признаков (TFIDF, n-grams, feature selection и др.). Полный список моделей доступен в документации существующего open source AutoML фреймворка FEDOT.

Такой подход позволяет сократить время и вычислительные ресурсы, необходимые для обучения, поскольку лучшие модели обучаются последовательно. Это приводит к быстрому нахождению наилучшего варианта, что на практике приводит к тому, что метрика качества оказывается гораздо выше, чем при базовом подходе с аналогичной сложностью.

С. Извлечение нейронных признаков

Предложенный подход достаточно гибок для объединения исходных данных с подготовленным вручную отображением набора данных в единое пространство признаков. Текстовые признаки отображаются с помощью предварительно обученного BERT-трансформатора. Категориальные и числовые признаки отображаются с помощью MLP. Все векторы конкатенируются и затем используются в качестве входных данных модели AutoML, как предложено в [13]. Эта модификация рассматривается как "Предлагаемый алгоритм" в разделе результатов и показана на рисунке 3. *prod_multimodal_source* обозначает готовый набор данных.

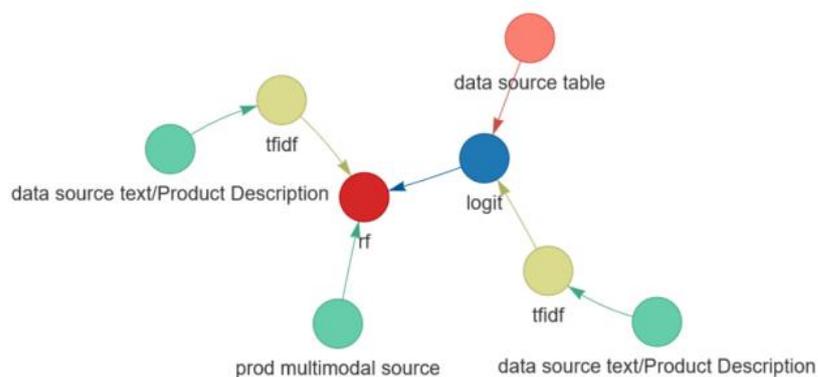


Рис. 3: Мультимодальный пайплайн, включающий нейронное извлечение признаков, сгенерированных с помощью предложенного подхода

V. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

Мы провели серию экспериментов, чтобы подтвердить корректность и эффективность предложенного подхода. В качестве эталонов использовались различные наборы данных для мультимодальной классификации и регрессии из AutoML Multimodal Benchmark [13]. Мы использовали сервер на базе Xeon Cascadelake (2900 МГц) с 12 ядрами и 24 ГБ системной памяти.

Методика эксперимента была разработана таким образом, чтобы обеспечить достоверность и воспроизводимость результатов. Изначально каждый набор данных был разделен на обучающий и валидационный подмножества в соотношении 75% к 25%.

Затем обучающий набор был предоставлен алгоритму эволюционной оптимизации. Для эволюционного алгоритма использовались следующие параметры: размер популяции - 20, количество поколений - 20. Для оптимизации процесса обучения использовался критерий раннего завершения: оптимизация прекращалась, если в течение 5 поколений не происходило улучшения метрики. На протяжении всего процесса оптимизации для расчета эффективности использования функций пригодности использовался метод 5-кратного перекрестного валидирования, что позволило получить надежную оценку прогностической способности модели.

Чтобы учесть стохастическую природу процесса эволюционной оптимизации, каждый эксперимент проводился 3 раза для всех наборов данных. Полученные показатели качества вычислялись как среднее значение по всем итерациям, что позволило получить исчерпывающую информацию о работе оптимизатора

A. Сравнение предложенного подхода с базовым

Первая серия экспериментов направлена на сравнение первоначальной реализации эволюционного AutoML с его мультимодальной модификацией. Результаты приведены в таблице I.

ТАБЛИЦА I: Сравнение базового и предлагаемого алгоритмов на различных наборах данных.

Dataset	Metric	Basic algorithm	Proposed algorithm	Metric gain, %
prod	Accuracy	0.872	0.885	+1.49
salary	Accuracy	0.368	0.382	+3.80
airbnb	Accuracy	0.401	0.415	+3.49
channel	Accuracy	0.504	0.526	+4.37
wine	Accuracy	0.703	0.756	+7.54
imdb	ROC-AUC	0.846	0.852	+0.71
fake	ROC-AUC	0.943	0.946	+0.32
kick	ROC-AUC	0.699	0.731	+4.58
jigsaw	ROC-AUC	0.925	0.994	+7.46
qaa	R2	0.443	0.486	+9.71
qaq	R2	0.426	0.477	+11.97
book	R2	0.842	0.885	+5.11
jc	R2	0.600	0.605	+0.83
cloth	R2	0.673	0.701	+4.16
ae	R2	0.891	0.952	+6.85
pop	R2	0.016	0.016	+0.00
house	R2	0.873	0.902	+3.32
mercari	R2	0.478	0.493	+3.14
average				+4.38

Как видно, предложенный алгоритм превосходит базовый по всем наборам данных, давая в среднем 4,38% выигрыша в метрике. Для каждого набора данных был проведен t-тест для двух независимых средних. Среднее значение р-значения составило 0,043, из чего можно сделать вывод о значимости результатов.

В. Сравнительный анализ предложенного алгоритма

В ходе следующего эксперимента мы сравнили предложенный подход с мультимодальным AutoML-фреймворком AutoGluon v0.8.0 [3]. В колонке предлагаемого подхода используется нейронный подход к извлечению признаков. Результаты приведены в таблице II. Для каждого набора данных был проведен t-тест между предложенным подходом и AutoGluon Multimodal для двух независимых значений. Среднее значение р равно 0,041. Можно сделать вывод, что результаты значимы.

ТАБЛИЦА II: Сравнение предложенного алгоритма с фреймворком AutoGluon

Dataset	Metric	AutoGluon	FEDOT	AutoGluon multimodal	Proposed approach	Metric gain, %
prod	Accuracy	0.895	0.897	0.909	0.921	+1.32
salary	Accuracy	0.528	0.402	0.496	0.436	-17.42
airbnb	Accuracy	0.466	0.434	0.459	0.445	-4.51
channel	Accuracy	0.550	0.542	0.556	0.542	-2.52
wine	Accuracy	0.842	0.784	0.836	0.812	-3.56
imdb	ROC-AUC	0.874	0.872	0.878	0.872	-0.68
fake	ROC-AUC	0.968	0.958	0.964	0.962	-0.62
kick	ROC-AUC	0.775	0.752	0.797	0.773	-3.01
jigsaw	ROC-AUC	0.915	0.998	0.967	0.998	+3.21
qaa	R2	0.383	0.504	0.438	0.524	+19.63
qaq	R2	0.426	0.554	0.456	0.586	+28.51
book	R2	0.600	0.994	0.638	0.997	+56.27
jc	R2	0.612	0.619	0.624	0.628	+0.64
cloth	R2	0.654	0.733	0.751	0.795	+5.86
ae	R2	0.979	0.974	0.977	0.976	-0.31
pop	R2	0.020	0.020	0.020	0.021	+5.00
house	R2	0.943	0.928	0.943	0.933	-1.06
mercari	R2	0.569	0.520	0.605	0.582	-3.80
average						+4.61

В результате предложенный подход не только улучшает среднюю метрику качества фреймворка FEDOT, но и превосходит более сложный подход, реализованный во фреймворке AutoGluon.

С. Анализ пайплайнов

Более высокое качество метрик в некоторых наборах данных можно объяснить с помощью анализа конечных пайплайнов. Как уже говорилось ранее, эволюционные алгоритмы имеют гораздо более широкое пространство поиска структур, чем стековые. На рисунке 4 показан конечный пайплайн для набора данных *prod*. Как видно, он имеет нетривиальную структуру с некоторыми интересными комбинациями моделей, такими как модели векторизации на основе TF IDF, балансировки выборки, классификатор случайного дерева, классификатор по методу Байеса.

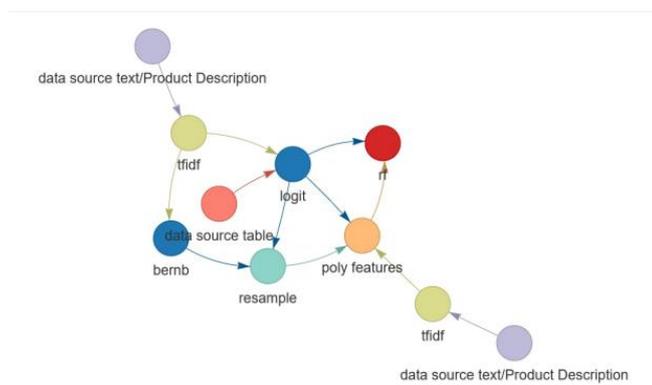


Рис. 4: Итоговый мультимодальный пайплайн на наборе данных *prod*

Д. Анализ эволюционной оптимизации

Чтобы доказать, что увеличение сложности пайплайна приводит к повышению метрики в наборе данных *prod*, мы провели эксперимент, представленный на рисунке 5, который визуализирует изменения показателя по поколениям. Показатель пригодности - это метрика, используемая для оценки производительности сгенерированных решений, в данном случае пайплайнов, в последовательных поколениях. Наблюдается четкая тенденция к росту пригодности по мере смены поколений, что говорит о том, что алгоритм эффективно оптимизирует пайплайны с течением времени. В первых поколениях наблюдается более широкий диапазон значений пригодности и большая вариативность, в то время как в последующих поколениях медианные значения пригодности выше, а вариативность меньше, что говорит о том, что алгоритм сходится к решениям с лучшими характеристиками.

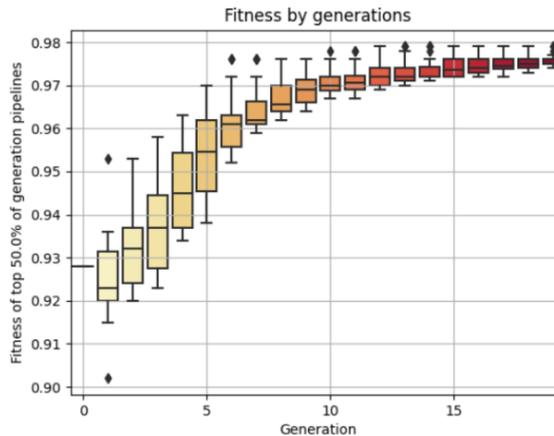


Рис. 5: Зависимость среднего значения метрики качества от номера поколения при оптимизации

На рисунке 6 показана доля различных операций, включенных во все сгенерированные пайплайны в разных поколениях процесса оптимизации. Ось x представляет собой номер поколения, указывающий на прогресс алгоритма оптимизации, а ось y показывает долю (или процент) всех генерируемых пайплайнов, включающих определенную операцию. Каждая цветная область представляет собой отдельную операцию, включенную в пайплайн. Операции перечислены в легенде в правой части графика, и каждый цвет соответствует набору операций, таких как: линейные, нелинейные, масштабирование признаков, вменения, инженерия признаков, кодирование, фильтрация и операции баланса данных. По мере развития поколений на графике видно, как меняется состав пайплайнов. Например, нелинейные операции вытесняют линейные, поскольку алгоритм оптимизации находит эти операции более подходящими для данного набора данных.

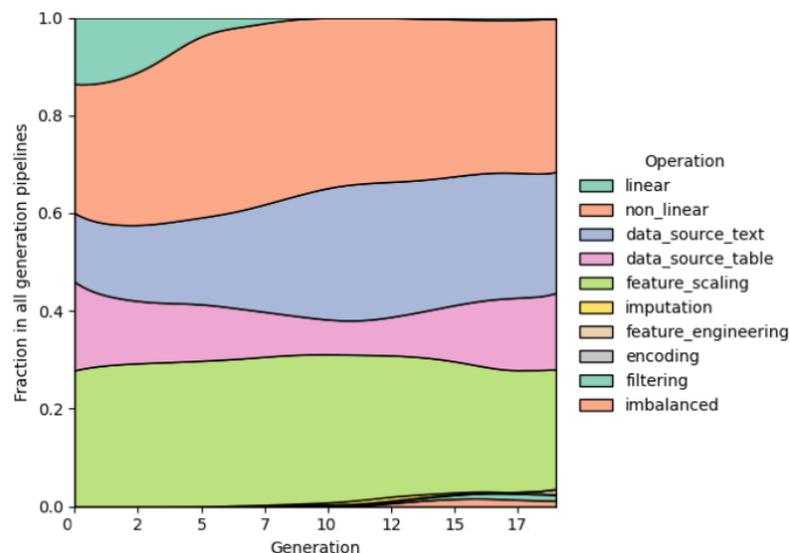


Рис. 6: Зависимость доли различных операций в пайплайнах от числа генераций при оптимизации

VI. ЗАКЛЮЧЕНИЕ

В данной статье мы предлагаем эволюционный подход для проектирования легких мультимодальных пайплайнов. Он реализован в рамках существующего открытого AutoML-фреймворка FEDOT и позволяет повысить эффективность моделирования для задач классификации и регрессии.

Эксперименты на известных бенчмарках подтверждают, что предложенный подход совместим с современными решениями и даже позволяет улучшить метрику качества почти на 4%. Полученные конвейеры относительно просты и могут быть проанализированы пошагово, в отличие от сложных сквозных решений, основанных на глубоком обучении.

Ограничения предложенного подхода заключаются в следующем:

- Нейронные методы извлечения признаков, используемые в данной работе, могут быть улучшены с помощью новых подходов (т.е. фундаментных моделей);
- Предложенный подход может решать только задачи классификации и регрессии. Возможно, он может быть расширен для решения более сложных задач, таких как распознавание именованных сущностей или семантическое сопоставление.

VII. ДОСТУПНОСТЬ ДАННЫХ И КОДА

Программная реализация предложенных алгоритмов и скрипты для проведенных экспериментов доступны в открытом репозитории <https://github.com/ITMO-NSS-team/fedot-multi-modal-paper>. Фреймворк FEDOT доступен в <https://github.com/aimclub/FEDOT>.

БЛАГОДАРНОСТЬ

Работа выполнена при поддержке Аналитического центра при Правительстве Российской Федерации (ИГК 000000D730324P540002), соглашение № 70-2021-00141.

ССЫЛКИ

[1] Т. Торнеде, А. Торнеде, Ж. Ханселле, М. Вевер, Ф. Мор и Э. Халлермайер, "На пути к экологичному автоматизированному машинному обучению: Статус кво и будущие направления," arXiv preprint arXiv:2111.05850, 2021.

[2] М. Фойрер, К. Эггенспергер, С. Фалькнер, М. Линдауэр и Ф. Хуттер, "Автоматическое обучение 2.0: Следующее поколение", CoRR, том abs/2007.04074, 2020. [Онлайн]. Доступно: <https://arxiv.org/abs/2007.04074>

[3] Н. Эрикссон, Дж. Мюллер, А. Ширков, Х. Чжан, П. Ларрой, М. Ли, и. А. Смола, "Autogluon-tabular: Надежный и точный автомат для структурированных данных," arXiv препринт arXiv:2003.06505, 2020.

[4] Х. Хэ, К. Чжао и Х. Чу, ""Автомл: Обзор последних достижений". Системы, основанные на знаниях, том 212, стр. 106622, 2021.

- [5] М.-А. Золлер и М. Ф. Хубер, "Бенчмарк и обзор автоматизированных фреймворков машинного обучения", препринт arXiv:1904.12054, 2019.
- [6] Н. О. Никитин, П. Вычужанин, М. Сарафанов, И. С. Полонская, И. Ревин, И. В. Барабанова, Г. Максимов, А. В. Калюжная, и А. Бухановский, "Автоматизированный эволюционный подход к проектированию композитных конвейеров машинного обучения", Компьютерные системы будущего поколения, том 127, стр. 109-125, 2022.
- [7] Н. Джин, Ф. Шолле, К. Сонг и Х. Ху, "Autokeras: Библиотека автоматов для глубокого обучения", Journal of Machine Learning Research, vol. 24, no. 6, pp. 1-6, 2023.
- [8] А. Вахрушев, А. Рыжков, М. Савченко, Д. Симаков, Р. Дамдинов, и А. Тужилин, ""Лайтавтомл": Автоматическое решение для крупной экосистемы финансовых услуг", препринт arXiv:2109.01528, 2021.
- [9] Э. ЛеДелл и С. Пуарье, "H2o automl: Масштабируемое автоматическое машинное обучение", в Трудах семинара AutoML на ICML, том 2020, 2020.
- [10] Р. С. Олсон и Ж. Н. Мура, ""Трот: Древовидная оптимизация конвейера для автоматизации машинного обучения", в семинаре по автоматическому машинному обучению. PMLR, 2016, pp. 66-74.
- [11] Н. W. Цзянь Ян, Сюэфэн Ли, "Гиперсети: Общая автоматизированная система машинного обучения ", <https://github.com/DataCanvasIO/Hypernets>, 2020, версия 0.2.x.
- [12] А. Мумуни и Ф. Мумуни, "Автоматизированная обработка данных и инженерия признаков для глубокого обучения и приложений больших данных: обзор", Журнал информации и интеллекта, 2024.
- [13] Х. Ши, Дж. Мюллер, Н. Эрикссон, М. Ли и А. Дж. Смола, "Бенчмаркинг мультимодальных автоматов для табличных данных с текстовыми полями", CoRR, vol. abs/2111.02705, 2021. [Онлайн]. Available: <https://arxiv.org/abs/2111.02705>
- [14] --, " Бенчмаркинг мультимодальных автоматов для табличных данных с текстовыми полями", arXiv preprint arXiv:2111.02705, 2021.
- [15] З. Танг, Н. Фанг, С. Чжоу, Т. Янг, З. Чжун, Т. Ху, К. Кирххофф и Г. Карыпис, "Autogluon-мультимодальная (automm): Суперзарядка мультимодальных автоматов с моделями оснований", препринт arXiv:2404.16233, 2024.
- [16] Ж. Аревало, Т. Солорио, М. М. у Гомес и Ф. А. Гонс алес, "Мультимодальные" блоки с управлением для слияния информации", 2017.

[17] Б. Скрля, М. Бевек и Н. Лавра, " Мультиmodalный автомат через эволюцию представлений", Машинное обучение и извлечение знаний, том 5, № 1, стр. 1-13, 2023. [Онлайн]. Доступно: <https://www.mdpi.com/2504-4990/5/1/1>

[18] Б. Скрля, М. Бевек и Н. Лавра, " Мультиmodalные автоматы через эволюцию представлений", Машинное обучение и извлечение знаний, том 5, № 1, стр. 1-13, 2022.

[19] Н. О. Никитин, П. Вычужанин, М. Сарафанов, И. С. Полонская, И. Ревин, И. В. Барабанова, Г. Максимов, А. В. Калюжная и А. Бухановский, "Автоматизированный эволюционный подход для проектирования составных конвейеров машинного обучения", Компьютерные системы будущего поколения, том 127, стр. 109-125, февраль 2022 г. [Онлайн]. Доступно: <http://dx.doi.org/10.1016/j.future.2021.08.022>

[20] У. Лю, У. Сунь, В. Сюэ, М. Чжан и Г. Г. Йен, "Обзор эволюционного поиска нейронных архитектур", CoRR, vol. abs/2008.10937, 2020. [Онлайн]. Доступно: <https://arxiv.org/abs/2008.10937>

[21] А. Рэдфорд, Ж. В. Ким, С. Халлейси, А. Рамеш, Г. Го, С. Агарвал, Г. Састри, А. Аскелл, Р. Мишкин, Ж. Кларк, Ж. Крюгер и И. Суцкевер, ""Обучение переносимых визуальных моделей на основе наблюдения за естественным языком", CoRR, vol. abs/2103.00020, 2021. [Онлайн]. Доступно: <https://arxiv.org/abs/2103.00020>

[22] И. С. Полонская, Н. О. Никитин, И. Ревин, П. Вычужанин и А. В. Калюжная, "Многоцелевое эволюционное проектирование составных моделей, управляемых данными", в 2021 IEEE Конгресс по эволюционным вычислениям (СЕС), 2021, стр. 926-933.