

УДК 004.89

DOI: [10.26102/2310-6018/2025.49.2.032](https://doi.org/10.26102/2310-6018/2025.49.2.032)

Автоматизированное проектирование интерпретируемой модели машинного обучения для оперативного прогнозирования силы ветра на морском побережье

П.А. Шерстнев¹, Е.С. Семенкин¹✉, С.А. Митрофанов¹, Т.Д. Ганчев²

¹*Сибирский государственный университет науки и технологий имени академика М.Ф. Решетнева, Красноярск, Российская Федерация*

²*Технический университет – Варна, Варна, Республика Болгария*

Резюме. В статье рассматривается задача проектирования системы оперативного краткосрочного прогнозирования скорости ветра в конкретной точке морского побережья. Предложен автоматизированный подход к проектированию гибридных моделей машинного обучения, объединяющих ансамбль многослойных нейронных сетей и интерпретируемую систему на нечеткой логике. Метод основан на автоматизированном формировании ансамбля нейронных сетей и системы на нечеткой логике с применением самоконфигурируемых эволюционных алгоритмов, что позволяет адаптироваться к особенностям входных данных без ручной настройки. После построения нейросетевого ансамбля формируется отдельная система на нечеткой логике, обучающаяся на его входах и выходах. Такой подход позволяет воспроизвести поведение нейросетевой модели в интерпретируемом виде. На основе экспериментальной проверки на метеорологическом датасете доказана эффективность метода, который обеспечивает баланс между качеством прогноза и интерпретируемостью модели. Показано, что построенная интерпретируемая система воспроизводит ключевые закономерности работы ансамбля нейросетей, оставаясь при этом компактной и понятной для анализа. Построенная модель может быть использована при принятии решений в портовых службах и при организации прибрежных мероприятий для быстрого и простого получения прогноза. Предложенный подход в целом позволяет получать аналогичные модели в различных ситуациях, похожих на рассмотренную.

Ключевые слова: оперативное прогнозирование ветровых характеристик, ансамбли нейронных сетей, системы на нечеткой логике, деревья решений, самоконфигурируемые эволюционные алгоритмы.

Для цитирования: Шерстнев П.А., Семенкин Е.С., Митрофанов С.А., Ганчев Т.Д. Автоматизированное проектирование интерпретируемой модели машинного обучения для оперативного прогнозирования силы ветра на морском побережье. *Моделирование, оптимизация и информационные технологии*. 2025;13(2). URL: <https://moitvvt.ru/ru/journal/pdf?id=1945> DOI: 10.26102/2310-6018/2025.49.2.032

Automated design of an interpretable machine learning model for operational coastal wind forecasting

P.A. Sherstnev¹, E.S. Semenkin¹✉, S.A. Mitrofanov¹, T.D. Ganchev²

¹*Reshetnev Siberian State University of Science and Technology, Krasnoyarsk, the Russian Federation*

²*Technical University of Varna, Varna, the Republic of Bulgaria*

Abstract. The article considers the problem of designing a system for operational short-term forecasting of wind speed at a specific point on the coast. An automated approach to designing hybrid machine learning models that combine an ensemble of multilayer neural networks and an interpretable system

based on fuzzy logic is proposed. The method is based on the automated formation of an ensemble of neural networks and a system based on fuzzy logic using self-configuring evolutionary algorithms, which allows adapting to the features of the input data without manual tuning. After constructing the neural network ensemble, a separate system based on fuzzy logic is formed, learning from its inputs and outputs. This approach allows reproducing the behavior of the neural network model in an interpretable form. Based on experimental testing on a meteorological dataset, the effectiveness of the method is proven, which ensures a balance between the quality of the forecast and the interpretability of the model. It is shown that the constructed interpretable system reproduces the key patterns of the neural network ensemble, while remaining compact and understandable for analysis. The constructed model can be used in decision-making in port services and in organizing coastal events for quick and easy forecasting. The proposed approach as a whole allows obtaining similar models in various situations similar to the one considered.

Keywords: operational forecasting of wind characteristics, ensembles of neural networks, fuzzy logic systems, decision trees, self-configuring evolutionary algorithms.

For citation: Sherstnev P.A., Semenkin E.S., Mitrofanov S.A., Ganchev T.D. Automated design of an interpretable machine learning model for operational coastal wind forecasting. *Modeling, Optimization and Information Technology*. 2025;13(2). (In Russ.). URL: <https://moitvvt.ru/ru/journal/pdf?id=1945> DOI: 10.26102/2310-6018/2025.49.2.032

Введение

Методы построения метеопрогнозов, использующие строго обоснованные математические модели, играют важную роль в предсказании различных аспектов погоды, включая условия на морских побережьях. Эти методы обычно описывают сразу несколько аспектов погоды (в зоне морского побережья обычно прогнозируют уровень волн, вызванных ветром определенной силы [1, 2]), охватывают достаточно большие территории (например, 5×5 км или больше) и предоставляют прогнозы на несколько дней вперед с последующими уточнениями. Это делает такие прогнозы менее подходящими для задач оперативного прогнозирования, требующих высокой точности и скорости реакции персонала.

Проблема оперативного прогнозирования в условиях морского побережья, когда особенно актуально достаточно точное прогнозирование силы ветра на временном горизонте 1–3 часа связана с необходимостью получения более точных данных для конкретных точек, а не общих прогнозов для больших площадей [3]. Существующие методы прогнозирования требуют определенных вычислительных ресурсов и времени для обработки данных, что может быть неприемлемо в ситуациях, когда информация нужна практически немедленно. Доступность достаточно точной и оперативной системы прогнозирования является критически важной для работников портовых служб и организаторов мероприятий на берегу моря [4]. Возможность получать актуальные данные здесь, сейчас и «на ладони» значительно повышает эффективность принятия решений и безопасность операций.

Еще одним требованием является обеспечение доступности системы. Требуется разработка интерфейса для мобильных устройств, обеспечивающего использование системы прогнозирования в любое время и в любом месте, безотносительно доступности сети Интернет и серьезных вычислительных ресурсов, так как это может быть критично для пользователей, особенно если появляются признаки того, что обычный предварительный прогноз недостаточно точно предсказывает погоду. Разрабатываемые в последнее время подходы с использованием моделей машинного обучения обычно тоже не удовлетворяют этим требованиям, так как используют глобальную, но не всем доступную информацию [5] и основываются на хотя и эффективных, но произвольно

выбранных методах [6] в результате чего получаются сложные неинтерпретируемые системы, требующие профессиональных знаний при их применении на практике.

Данная статья посвящена автоматизации разработки и реализации простой, но достаточно точной и при этом интерпретируемой модели машинного обучения для оперативного прогнозирования силы ветра на морском побережье.

Основные аспекты разработки модели включают:

- автоматизацию процесса проектирования модели: использование самоконфигурируемых эволюционных алгоритмов, позволяющих автоматически генерировать структуру и подбирать параметры модели на основе статистических данных за короткий промежуток времени, что исключает влияние личных предпочтений разработчика;

- интерпретируемость модели: важно не только получить точный прогноз, но и обеспечить возможность его интерпретации, позволяющей конечным пользователям лучше понимать причины изменений в прогнозе и принимать более обоснованные решения;

- использование для прогнозирования статистических данных за небольшой промежуток времени, например, за предыдущий час, что позволит пользователю оперативнее подгружать данные и делать более точные предсказания развития ситуации;

- доступность системы – разработка интерфейса для мобильных устройств или веб-приложений обеспечит доступ к системе в любое время и в любом месте, что может быть критично для конечных пользователей.

Используемый подход

Современные достижения в области интеллектуальных информационных технологий (ИИТ) открывают новые возможности для решения сложных задач анализа данных, однако одновременно усложняются и применяемые модели машинного обучения (МО). Это приводит к снижению их интерпретируемости и возникновению феномена «черного ящика», при котором внутренние механизмы принятия решений становятся недоступными для анализа. Сложные системы, такие как многослойные искусственные нейронные сети (ИНС) [7], демонстрируют достаточно высокую точность прогнозов, но отсутствие прозрачности значительно ограничивает область их применения.

В противоположность этому существует направление интерпретируемых моделей МО, примерами которых являются нечеткие логические системы (НЛС) [8] и деревья решений (ДР) [9]. НЛС и ДР обеспечивают интерпретируемость решений за счет использования определенных логических правил, позволяющих явно описывать зависимости между входными признаками и выходными результатами, однако зачастую уступают ИНС в точности предсказаний. Еще одной возможной проблемой может быть размер и сложность этих моделей (количество и длина правил НЛС, глубина ДР, и т. д.), что служит препятствием для их понимания и интерпретации человеком.

В связи с этим актуальной задачей становится объединение преимуществ сложных высокоточных моделей и интерпретируемых систем. В рамках данного подхода ансамбль ИНС используется для решения основной прогностической задачи, в то время как НЛС строится для интерпретации поведения ансамбля, объясняя процессы преобразования входных данных в выходные решения. Процесс автоматизированного построения как ансамблей нейросетей, так и интерпретирующих их нечетких систем представляет собой сложную оптимизационную задачу, требующую эффективного поиска в многомерных пространствах смешанных переменных. Аналогичные трудности возникают и при автоматизированном проектировании деревьев решений.

Эволюционные алгоритмы (ЭА) доказали свою эффективность в решении подобных задач благодаря способности к одновременному структурному и параметрическому поиску, минимизируя необходимость ручной настройки [10]. Таким образом, автоматизация проектирования ансамблей ИНС и сопровождающих их интерпретируемых НЛС, а также ДР с использованием ЭА представляет собой перспективное направление развития ИИТ.

Автоматизированное проектирование интеллектуальных систем эволюционными алгоритмами

Ансамбли нейронных сетей. Эффективность НС во многом определяется ее архитектурой – числом и связями нейронов, количеством скрытых слоев, а также типами используемых функций активации. Подбор оптимальной структуры может быть выполнен с помощью ЭА без необходимости полного перебора вариантов или сложного анализа данных. Так, в [11] предложен метод оптимизации структуры НС с помощью метода генетического программирования (ГП) [12, 13], обеспечивающий автоматизированный поиск архитектур с учетом эффективности их работы на обучающих данных. Архитектуры нейронных сетей кодируются в виде бинарных деревьев. Узлы дерева делятся на функциональные (+, >) и терминальные элементы (in1, in2, 2gs и др.), определяющие операции объединения нейронов и их последовательного соединения в слои. Пример кодирования одиночной нейронной сети представлен на Рисунке 1.

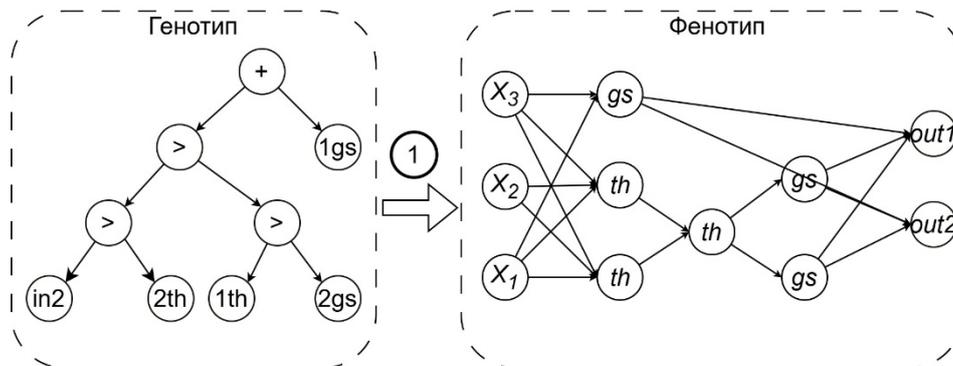


Рисунок 1 – Пример кодирования нейронной сети в виде дерева
 Figure 1 – Example of neural network coding in the form of a tree

Оптимизация структуры сетей осуществляется методом ГП, где функция пригодности определяется ошибкой НС на обучающем наборе данных. Поскольку сети могут иметь произвольную архитектуру, настройка весов осуществляется с применением ЭА.

Для формирования ансамблей в рамках одного дерева применяется модификация кодирования: вводятся комбинированные узлы, которые одновременно выполняют роль блока нейронов и задают тип соединения. Это позволяет описывать в одном дереве несколько нейросетей с различной архитектурой и активациями. Для объединения прогнозов отдельных сетей используется простое усреднение, голосование или стекинг. В стекинге мета-модель реализуется в виде однослойного перцептрона, обучающегося на выходах сетей-участников (Рисунок 2). Для повышения разнообразия ансамбля используется бутстрап-процедура формирования обучающих подвыборок для каждой нейронной сети, что позволяет снизить корреляцию между участниками ансамбля и повысить обобщающую способность итоговой модели.

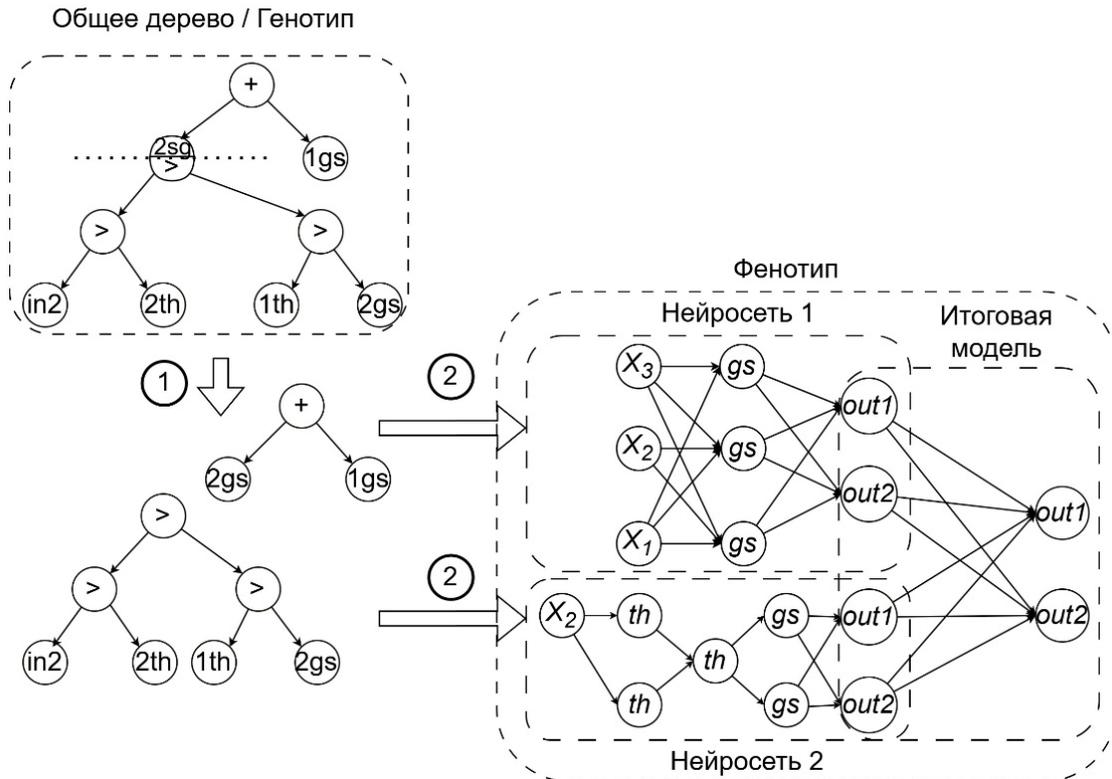


Рисунок 2 – Стекинг-ансамбль нейронных сетей
 Figure 2 – Stacking ensemble of neural networks

Дополнительно возможно расширение подхода через использование двух бинарных деревьев: первое кодирует архитектуры базовых сетей, второе – структуру мета-модели стекинга.

Системы на нечеткой логике. Формирование НЛС сводится к задаче выбора оптимальной базы правил из множества возможных комбинаций. Поскольку количество возможных баз правил экспоненциально увеличивается с ростом числа входных признаков и термов, перебор всех вариантов становится вычислительно невозможным. В связи с этим для эффективного поиска оптимальной базы правил целесообразно применять эволюционные алгоритмы (ЭА) [14]. Для применения ЭА необходимо закодировать возможные решения в форме, пригодной для операций оптимизации. В рамках настоящего исследования используется питсбургский подход кодирования, при котором одним индивидом в популяции является целая база правил [15]. Каждое правило представляет собой набор предпосылок, связанных логическим оператором "И", и заключения. Структура индивидуального кодирования определяется следующей длиной бинарной строки:

$$1 + \lceil \log_2(n_{sets} + 1) \rceil \cdot n_{var} + \lceil \log_2(n_{outsets}) \rceil \cdot n_{rules},$$

где n_{sets} – количество нечетких термов на входных переменных, n_{var} – количество входных переменных, $n_{outsets}$ – количество термов на каждой выходной переменной, $n_{ruleset}$ – максимальное количество правил в базе.

Дополнительно для каждого правила вводится управляющий бит, определяющий его использование в базе, а также специальный терм игнорирования, позволяющий адаптивно управлять длиной правил и объемом базы. При таком способе кодирования особенно эффективно применение генетического алгоритма (ГА) [16, 17], так как он позволяет проводить эволюционный поиск в пространстве решений, представленном в

виде бинарных строк. Это упрощает реализацию операторов мутации и скрещивания, а также способствует эффективному исследованию сложных дискретных пространств признаков.

Функция пригодности отражает качество предсказаний, обеспечиваемое нечетко-логической системой при использовании данной базы правил, с учетом штрафов за увеличение количества и длины правил. В зависимости от типа выходных данных: при дискретных выходах используется схема классификации с присвоением меток классов; при непрерывных выходах применяются методы нечеткого логического вывода, такие как Мамдани, для аппроксимации выходных значений [18].

При применении эволюционных алгоритмов для автоматизированного формирования ИИТ, включающих ансамбли нейронных сетей и нечетко-логические системы, возникает проблема выбора эффективных операторов эволюции (селекции, мутации, скрещивания). В настоящей работе применяется метод самоконфигурирования [19], основанный на адаптивной корректировке вероятностей применения операторов эволюции в зависимости от их вклада в улучшение пригодности особей на текущем поколении.

Деревья принятия решений. Одним из альтернативных подходов к построению интерпретируемых моделей машинного обучения является использование деревьев решений [20]. Дерево решений представляет собой бинарную структуру, где внутренние узлы содержат функции, а листья дерева – предсказания. Чаще всего используются одномерные предикаты, сравнивающие значение отдельного признака с порогом, но также существуют многомерные предикаты. Хотя они позволяют создавать более сложные разделяющие поверхности, их редко применяют из-за повышенной склонности модели к переобучению [21].

При построении деревьев решений задается функционал качества, на основе которого осуществляется разбиение выборки в каждой вершине. Далее используется функционал следующего вида:

$$Q(R_m, j, s) = H(R_m) - \frac{|R_l|}{|R_m|} H(R_l) - \frac{|R_r|}{|R_m|} H(R_r),$$

где $H(R)$ – это критерий информативности, который оценивает качество распределения целевой переменной среди объектов множества R , R_m – множество объектов, попавших в некоторую вершину, R_l и R_r – объекты, попадающие в левое и правое поддерево, соответственно, при заданном предикате.

Ключевой особенностью функционала качества при решении задачи регрессии является определение критерия информативности через среднюю квадратичную ошибку. Чем меньше разнообразие целевой переменной, тем меньше должно быть значение критерия информативности и, соответственно, его значение минимизируется. Для автоматизации решения задачи с помощью деревьев решений будем использовать самоконфигурируемый алгоритм генетического программирования [22] для выбора структуры и модифицированный процесс обучения [23], предполагающий двухэтапную оптимизацию критерия информативности $H(R)$.

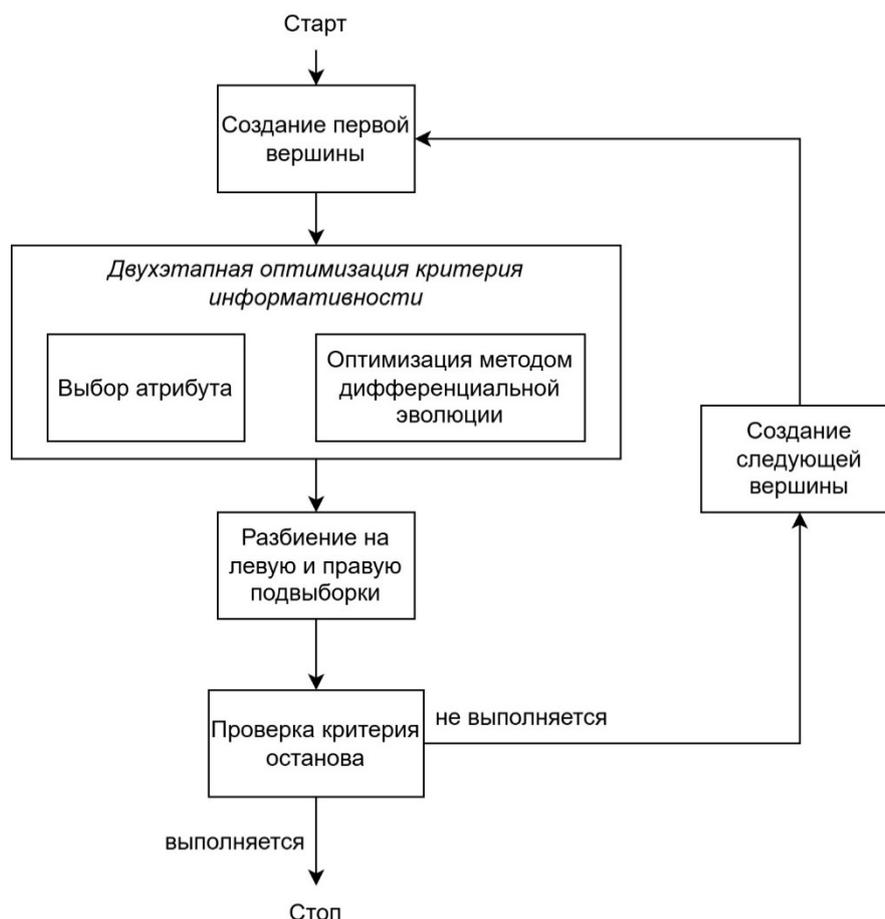


Рисунок 3 – Модифицированный алгоритм обучения дерева решений с применением двухэтапной оптимизации критерия информативности

Figure 3 – Modified decision tree learning algorithm using two-stage optimization of the information criterion

На Рисунке 3 представлена модифицированная схема алгоритма обучения дерева принятия решений, куда включены шаги двухэтапной оптимизации. На первом этапе оптимизируется процесс выбора атрибута разделения, а затем для него оптимизируется пороговое значение. Выбор атрибута разделения осуществляется на основе корреляции атрибута и целевого значения. Чем выше корреляция, тем большую значимость несет атрибут, и, как следствие, он и выбирается. На втором этапе предполагается оптимизация порогового значения. Для оптимизации порогового значения выбранного атрибута используется метод дифференциальной эволюции [24], один из эволюционных методов, предназначенный для решения задачи многомерной оптимизации. Метод использует идеи генетических алгоритмов, но, в отличие от них, не требует работы с переменными в бинарном коде.

Взаимодействие ансамбля нейронных сетей и нечеткой логической системы

Используется подход к повышению интерпретируемости решений, принимаемых ансамблем нейронных сетей (АНС), путем построения эквивалентной НЛС [25]. Как АНС, так и НЛС в данном подходе проектируются автоматически с использованием самоконфигурируемых ЭА, однако с разной целевой направленностью: АНС обучается на достижение высокой точности моделирования, тогда как НЛС строится для воспроизведения поведения АНС с сохранением интерпретируемости. Полученная база

правил НЛС должна объяснять, каким образом АНС преобразует входные данные в выходные результаты. Для обеспечения интерпретации необходимо, чтобы выходы НЛС максимально точно соответствовали выходам АНС при одинаковых входных данных.

Обобщенный алгоритм взаимодействия АНС и НЛС включает следующие этапы:

1. Проектирование АНС: автоматизированное построение эффективных структур и обучение АНС на исходной обучающей выборке (X, Y) , где $X \subseteq \mathbb{R}^n$ – пространство признаков, а $Y \subseteq \mathbb{R}^m$ – множество целевых векторов выходов (в случае непрерывных задач).

2. Генерация обучающей выборки для НЛС: получение входов X_{NN} и соответствующих выходов Y_{NN} АНС на данных задачи.

3. Проектирование нечетко-логической системы: автоматизированное формирование базы правил НЛС на основе данных (X_{NN}, Y_{NN}) , с целью воспроизведения поведения АНС.

4. Анализ интерпретируемости: исследование сформированной базы правил для выявления закономерностей преобразования признаков во множество выходных параметров.

Таким образом, в результате процедуры получается интерпретируемая НЛС, способная объяснять работу АНС на множестве входных данных.

Постановка задачи прогнозирования

Для построения моделей прогнозирования использовался метеорологический датасет WindFc¹, содержащий данные о ветровых характеристиках и погодных условиях. Набор данных включает измерения, зафиксированные в течение одного календарного года с частотой один раз в минуту. Впоследствии данные были агрегированы с интервалом в 15 минут для приведения к структуре прогнозируемых значений.

Целью прогнозирования является предсказание скорости ветра (WindSpeed) на горизонт 3 часа вперед. Прогноз осуществляется с временным разрешением в 15 минут, что соответствует необходимости формировать 12 предсказаний относительно каждого текущего момента времени. Основной фокус ставится на переменной WindSpeed (средняя скорость ветра).

На этапе подготовки данных был проведен анализ корреляций между признаками. Результаты показали, что значимую связь с целевой переменной имеют только скорость ветра и максимальная скорость ветра, тогда как температура воздуха, атмосферное давление, влажность и направление ветра продемонстрировали слабую корреляцию и были исключены. С учетом цели построения интерпретируемых моделей было принято решение оставить в качестве признаков только эти два параметра.

Для учета временных зависимостей был проведен анализ автокорреляционных и частных автокорреляционных функций (ACF и PACF). На основании полученных результатов в модель были включены три лага предыдущих значений признаков. Это позволило учитывать краткосрочные динамические эффекты без чрезмерного увеличения сложности модели.

На этапе подготовки данных признаки и целевые переменные были сформированы по принципу скользящего окна: для каждой временной точки использовались значения скорости ветра и порывов за три предыдущих интервала и текущий момент времени. Целевыми переменными являются прогнозные значения скорости ветра на горизонте 3 часа вперед с шагом 15 минут.

¹ Resources. Artificial Intelligence Laboratory. URL: <http://ailab.tu-varna.bg/index.php/resources> (дата обращения: 05.05.2025).

Итоговый набор данных после обработки включает 31521 обучающих и 3490 тестовых наблюдений для варианта с одним часом истории. Пример структуры сформированных входных признаков и выходных переменных приведен в Таблице 1.

Таблица 1 – Описание входов и выходов для задачи краткосрочного прогноза ветра
Table 1 – Description of inputs and outputs for the short-term wind forecast problem

Тип	Переменные	Описание
Вход	WindSpeed (T-45), WindSpeed (T-30), WindSpeed (T-15), WindSpeed (T)	Значения скорости ветра за 3 предыдущих интервала и в текущий момент времени
Вход	WindSpeedMax (T-45), WindSpeedMax (T-30), WindSpeedMax (T-15), WindSpeedMax (T)	Значения порывов ветра за 3 предыдущих интервала и в текущий момент времени
Выход	WindSpeed (T+15), (T+30), ..., (T+180)	Прогноз скорости ветра на горизонте от 15 до 180 минут вперёд

Описание эксперимента

Для обеспечения надежности результатов все методы запускались по 20 раз, после чего значения целевых метрик усреднялись. Это необходимо, поскольку применяемые алгоритмы моделирования и оптимизации являются стохастическими процедурами и могут демонстрировать вариативность результатов между запусками.

Формирование АНС осуществлялось с использованием алгоритма ГП с ограничением в 50 поколений при 20 индивидах в каждом поколении. Специфика данной задачи такова, что выделение большего вычислительного ресурса не имело смысла, так как это не приводило к увеличению точности, что было показано в ходе пробных запусков. Оптимизация весовых коэффициентов ансамбля проводилась с помощью алгоритма SHADE [26] с максимальным числом 5000 итераций, размером популяции 200 особей и применением ранней остановки при отсутствии улучшения в течение 100 итераций. Формирование НЛС осуществлялось с помощью ГА с размером популяции 200 особей и максимальным числом поколений 400.

В эксперименте оценивались шесть показателей, выраженных через среднеквадратичную ошибку (RMSE, Root Mean Square Error):

1. RMSE ансамбля нейронных сетей (АНС) при решении исходной задачи прогнозирования.
2. RMSE системы на нечеткой логике (НЛС), обученной на исходных данных (X, Y).
3. RMSE НЛС1, обученной на данных, сформированных по входам и выходам АНС (X_{NN} , Y_{NN});
4. RMSE НЛС2 из пункта 3, рассчитанная на исходной задаче прогнозирования.
5. RMSE стандартного дерева принятия решений (ДПР).
6. RMSE модифицированного дерева принятия решений (ДПРс).

Проверка статистической значимости различий между сравниваемыми методами осуществлялась с использованием критерия Манна-Уитни при уровне значимости $\alpha = 0,05$ [27].

Результаты вычислительного эксперимента

Усредненные по 20 независимым запускам результаты представлены в Таблице 2. Строки соответствуют прогнозируемым значениям скорости ветра на различных

горизонтах от 15 до 180 минут вперед; столбцы (1)–(6) содержат значения RMSE для соответствующих подходов.

Таблица 2 – Результаты моделирования
Table 2 – Simulation results

Выход	АНС	НЛС	НЛС1	НЛС2	ДПР	ДПРс
WindSpeed (T+15)	1,39	1,983	1,081	1,910	0,844	0,809
WindSpeed (T+30)	1,49	1,983	1,042	1,925	0,999	0,99
WindSpeed (T+45)	1,53	1,984	0,995	1,929	1,096	1,07
WindSpeed (T+60)	1,57	1,985	0,954	1,927	1,161	1,18
WindSpeed (T+75)	1,60	1,985	0,912	1,939	1,24	1,235
WindSpeed (T+90)	1,63	1,985	0,857	1,933	1,285	1,297
WindSpeed (T+105)	1,64	1,986	0,857	1,934	1,34	1,34
WindSpeed (T+120)	1,66	1,987	0,792	1,931	1,343	1,374
WindSpeed (T+135)	1,69	1,987	0,803	1,939	1,408	1,428
WindSpeed (T+150)	1,72	1,987	0,777	1,950	1,461	1,478
WindSpeed (T+165)	1,73	1,988	0,741	1,954	1,488	1,506
WindSpeed (T+180)	1,76	1,988	0,747	1,960	1,483	1,52
Среднее	1,62	1,975	0,880	1,936	1,26	1,27

НЛС1, обученная на входах и выходах АНС, показала среднюю ошибку 0,880 при воспроизведении выходов АНС. Это означает, что она с умеренной ошибкой аппроксимирует поведение ансамбля и может быть использована для интерпретации его решений. Однако численные показатели не представляют практического интереса, так как это точность аппроксимации выходов АНС, а не прогноза скорости ветра.

Лучшую точность на первый час прогноза демонстрирует ДПРс, а на более поздних сроках – ДПР, но средняя ошибка прогноза у них практически совпадает. Однако автоматически спроектированные деревья решений, демонстрирующие такие результаты, очень велики (более 40 уровней) и не являются интерпретируемыми. Тем не менее они могут быть полезными в ситуациях, когда небольшая разница в точности прогноза является главным свойством, а интерпретируемость модели оказывается вторичной.

НЛС, построенная ГА и обученная напрямую на исходных данных, дала ошибку – 1,975, что является неплохим результатом. Однако эта НЛС велика и ее правила содержат 8 предпосылок, что делает такую модель также практически не интерпретируемой.

АНС, построенный с использованием ГП, показал хорошее качество прогнозирования скорости ветра, обеспечив среднюю ошибку (RMSE) 1,62, однако это тоже не интерпретируемая модель.

Единственной интерпретируемой моделью оперативного прогноза ветра морского побережья оказывается НЛС2, дающая при ее применении к исходной задаче прогнозирования оценку ошибки 1,936, что лучше, чем у НЛС, построенной на исходных данных. Таким образом, передача поведения сложной модели через НЛС позволяет улучшить прогнозные свойства интерпретируемой модели. Дополнительно, при обучении НЛС на выходах АНС (НЛС2) удалось сократить среднее число предпосылок в базе правил с 8 до 5, что упрощает модель и делает ее интерпретируемой. В результате, построенная таким образом НЛС не только близка к поведению АНС, но и представляет собой более прозрачную и удобную для анализа и принятия решений систему.

Точность, достигнутая всеми моделями, является достаточной с точки зрения практики и соответствует точности прогнозов, получаемой в аналогичных целях другими, в том числе и намного более сложными, моделями машинного обучения [5, 6].

Ниже приведен пример одного из правил, сформированных НЛС на исходных данных. Данное правило содержит восемь предпосылок, что делает модель излишне объемной и неудобной для интерпретации. Правило описывает связь между предшествующими значениями метеопараметров (предпосылками) и будущими состояниями прогнозируемой переменной – скоростью ветра (заклЮчениями) на горизонте до 180 минут вперед с шагом 15 минут.

ПРАВИЛО 1: ЕСЛИ (WindSpeedMax[t-45мин] крайне низкое) И (WindSpeedMax[t-30мин] низкое) И (WindSpeedMax[t-15мин] очень низкое) И (WindSpeedMax[t] высокое) И (WindSpeed[t-45мин] очень низкое) И (WindSpeed[t-30мин] низкое) И (WindSpeed[t-15мин] очень низкое) И (WindSpeed[t] высокое)

ТОГДА WindSpeed[t+15мин] очень высокое, WindSpeed[t+30мин] высокое, WindSpeed[t+45мин] крайне низкое, WindSpeed[t+60мин] крайне высокое, WindSpeed[t+75мин] очень высокое, WindSpeed[t+90мин] низкое, WindSpeed[t+105мин] крайне низкое, WindSpeed[t+120мин] очень высокое, WindSpeed[t+135мин] крайне низкое, WindSpeed[t+150мин] крайне высокое, WindSpeed[t+165мин] очень низкое, WindSpeed[t+180мин] крайне низкое.

Далее представлена база из четырех нечетких правил, сформированная на основе поведения ансамбля нейросетей (АНС). Эта база была получена путем моделирования выходов АНС при помощи интерпретируемой модели – НЛС2.

ПРАВИЛО 1: ЕСЛИ (WindSpeedMax[t-45мин] среднее) И (WindSpeedMax[t-30мин] среднее) И (WindSpeedMax[t-15мин] среднее) И (WindSpeed[t-15мин] высокое) И (WindSpeed[t] среднее)

ТОГДА WindSpeed[t+15мин] высокое, WindSpeed[t+30мин] высокое, WindSpeed[t+45мин] среднее, WindSpeed[t+60мин] среднее, WindSpeed[t+75мин] высокое, WindSpeed[t+90мин] высокое, WindSpeed[t+105мин] низкое, WindSpeed[t+120мин] среднее, WindSpeed[t+135мин] среднее, WindSpeed[t+150мин] высокое, WindSpeed[t+165мин] низкое, WindSpeed[t+180мин] среднее;

ПРАВИЛО 2: ЕСЛИ (WindSpeedMax[t-45мин] высокое) И (WindSpeedMax[t-30мин] высокое) И (WindSpeedMax[t-15мин] высокое) И (WindSpeed[t-15мин] высокое) И (WindSpeed[t] высокое)

ТОГДА WindSpeed[t+15мин] высокое, WindSpeed[t+30мин] очень высокое, WindSpeed[t+45мин] высокое, WindSpeed[t+60мин] высокое, WindSpeed[t+75мин] очень высокое, WindSpeed[t+90мин] среднее, WindSpeed[t+105мин] высокое, WindSpeed[t+120мин] высокое, WindSpeed[t+135мин] среднее, WindSpeed[t+150мин] высокое, WindSpeed[t+165мин] высокое, WindSpeed[t+180мин] высокое;

ПРАВИЛО 3: ЕСЛИ ($WindSpeedMax[t-45мин]$ высокое) И ($WindSpeedMax[t-30мин]$ высокое) И ($WindSpeedMax[t-15мин]$ высокое) И ($WindSpeed[t-15мин]$ крайне высокое) И ($WindSpeed[t]$ очень высокое)

ТОГДА $WindSpeed[t+15мин]$ среднее, $WindSpeed[t+30мин]$ очень высокое, $WindSpeed[t+45мин]$ высокое, $WindSpeed[t+60мин]$ среднее, $WindSpeed[t+75мин]$ очень высокое, $WindSpeed[t+90мин]$ высокое, $WindSpeed[t+105мин]$ высокое, $WindSpeed[t+120мин]$ высокое, $WindSpeed[t+135мин]$ высокое, $WindSpeed[t+150мин]$ очень высокое, $WindSpeed[t+165мин]$ высокое, $WindSpeed[t+180мин]$ среднее;

ПРАВИЛО 4: ЕСЛИ ($WindSpeedMax[t-45мин]$ очень высокое) И ($WindSpeedMax[t-30мин]$ очень высокое) И ($WindSpeedMax[t-15мин]$ очень высокое) И ($WindSpeed[t-15мин]$ очень высокое) И ($WindSpeed[t]$ очень высокое)

ТОГДА $WindSpeed[t+15мин]$ крайне высокое, $WindSpeed[t+30мин]$ очень высокое, $WindSpeed[t+45мин]$ очень высокое, $WindSpeed[t+60мин]$ очень высокое, $WindSpeed[t+75мин]$ крайне высокое, $WindSpeed[t+90мин]$ крайне высокое, $WindSpeed[t+105мин]$ очень высокое, $WindSpeed[t+120мин]$ очень высокое, $WindSpeed[t+135мин]$ очень высокое, $WindSpeed[t+150мин]$ очень высокое, $WindSpeed[t+165мин]$ очень высокое, $WindSpeed[t+180мин]$ крайне высокое.

Каждое правило направлено на предсказание значений скорости ветра на 12 шагов вперед (от $t+15мин$ до $t+180мин$), что обуславливает наличие развернутой части заключений. Тем не менее, число предпосылок остается ограниченным – всего 4–5 условий на основе недавней истории параметров. Это свидетельствует о том, что модель может формировать комплексный прогноз на достаточно длительный интервал времени (3 часа), опираясь на компактное множество входных признаков. Подобное свойство делает модель интерпретируемой и пригодной для анализа и принятия решений, в отличие от «черного ящика» оригинального ансамбля и вычислительных формул деревьев решений, а также обычной НЛС.

Таким образом, представленная база правил одновременно отражает поведение исходной нейросетевой модели, упрощает структуру принятия решений за счет меньшего числа предпосылок, обеспечивает объяснимость и компактность описания прогнозируемого процесса. Если она будет реализована на переносимом устройстве, например, планшете, то может служить для поддержки принятия решений непосредственно персоналом морских портов и прибрежного обслуживания на месте выполнения им работы.

Заключение

Разработка эффективной модели машинного обучения для оперативного прогнозирования силы ветра на морском побережье представляет собой важный шаг к улучшению метеорологических услуг в данной области. Автоматизация процесса создания модели с использованием ЭА и акцент на простоту и интерпретируемость обеспечивают высокую точность прогноза и удобство использования системы. Это позволит работникам портовых служб и организаторам мероприятий на морском побережье принимать более обоснованные решения в условиях быстро меняющейся погоды, обеспечивая более высокий уровень безопасности.

В настоящем исследовании продемонстрирована возможность автоматизированного проектирования интерпретируемых моделей краткосрочного прогноза скорости ветра на морском побережье путем сочетания АНС и НЛС, проектируемых с помощью самоконфигурируемых ЭА. Полученные результаты подтвердили, что построенные НЛС, обученные на выходах ансамбля, позволяют воспроизвести его поведение, обеспечивая при этом высокую интерпретируемость.

Автоматически сформированная НЛС обеспечила приближение к прогнозным характеристикам АНС и сформировала более компактную базу правил, что существенно повышает прозрачность принятия решений. Таким образом, предложенный подход может рассматриваться как эффективный инструмент построения объяснимых моделей прогнозирования для сложных систем, сочетающий точность, интерпретируемость и адаптивность.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Ратнер Ю.Б., Фомин В.В., Иванчик А.М., Иванчик М.В. Система оперативного прогноза ветрового волнения Черноморского центра морских прогнозов. *Морской гидрофизический журнал*. 2017;(5):56–66. <https://doi.org/10.22449/0233-7584-2017-5-56-66>
Ratner Yu.B., Fomin V.V., Ivanchik A.M., Ivanchik M.V. System of the Wind Wave Operational Forecast by the Black Sea Marine Forecast Center. *Physical Oceanography*. 2017;(5):51–59. <https://doi.org/10.22449/1573-160X-2017-5-51-59>
2. Зеленько А.А., Струков Б.С., Реснянский Ю.Д., Мартынов С.Л. Система прогнозирования ветрового волнения в мировом океане и морях России. *Труды Государственного океанографического института*. 2014;(215):90–101.
Zelenko A.A., Strukov B.S., Resnyansky Yu.D., Martynov S.L. A System of Wind Wave Forecasting in the World Ocean and Seas of Russia. *Trudy Gosudarstvennogo okeanograficheskogo instituta*. 2014;(215):90–101. (In Russ.).
3. Solari G., Repetto M.P., Burlando M., et al. The Wind Forecast for Safety Management of Port Areas. *Journal of Wind Engineering and Industrial Aerodynamics*. 2012;104–106:266–277. <https://doi.org/10.1016/j.jweia.2012.03.029>
4. Burlando M., De Gaetano P., Pizzo M., Repetto M.P., Solari G., Tizzi M. Wind Short-Term Forecast in Port Areas. In: *Proceedings of the 6th European-African Conference on Wind Engineering (EACWE), 07–11 July 2013, Cambridge, United Kingdom*. 2013. P. 6–8.
5. Chu X., Bai W., Sun Yu., Li W., Liu C., Song H. A Machine Learning-Based Method for Wind Fields Forecasting Utilizing GNSS Radio Occultation Data. *IEEE Access*. 2022;10:30258–30273. <https://doi.org/10.1109/ACCESS.2022.3159231>
6. Zhang W., Tian M., Hai Sh., et al. Improving the Forecasts of Coastal Wind Speeds in Tianjin, China Based on the WRF Model with Machine Learning Algorithms. *Journal of Meteorological Research*. 2024;38(3):570–585. <https://doi.org/10.1007/s13351-024-3096-z>
7. Аксенов С.В., Новосельцев В.Б. *Организация и использование нейронных сетей (методы и технологии)*. Томск: Изд-во НТЛ; 2006. 128 с.
8. Тэрано Т., Асаи К., Сугэно М. *Прикладные нечеткие системы*. Москва: Мир; 1993. 368 с.
9. Rokach L., Maimon O.Z. *Data Mining with Decision Trees: Theory and Applications*. Singapore: World Scientific Publishing Co. Pte. Ltd.; 2008. 244 p.
10. Semenkin E. Computational Intelligence Algorithm-Based Comprehensive Human Expert and Data-Driven Model Mining for the Control, Optimization and Design of Complicated Systems. *International Journal on Information Technologies & Security*. 2019;11:63–66.
11. Липинский Л.В., Семенкин Е.С. Применение алгоритма генетического программирования в задачах автоматизации проектирования интеллектуальных информационных технологий. *Вестник Сибирского государственного аэрокосмического университета им. академика М.Ф. Решетнева*. 2006;(3):22–26.

- Lipinsky L.V., Semenkin E.S. Application of Genetic Programming Algorithm in Automated Design of Intellectual Information Technologies. *Vestnik SibSAU. Aerospace Technologies and Control Systems*. 2006;(3):22–26. (In Russ.).
12. Koza J.R. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge: The MIT Press; 1998. 819 p.
 13. Al-Helali B., Chen Q., Xue B., Zhang M. Genetic Programming-Based Feature Selection for Symbolic Regression on Incomplete Data. *Evolutionary Computation*. 2024;1–27. https://doi.org/10.1162/evco_a_00362
 14. Gorzalczany M.B., Rudziński F. Time-Series-Dynamics Modelling and Forecasting – An Accurate and Interpretable Genetic-Fuzzy Approach. In: *Advances in Fuzzy Logic and Technology 2017: Proceedings of: EUSFLAT-2017 – The 10th Conference of the European Society for Fuzzy Logic and Technology, 11–15 September 2017, Warsaw, Poland; IWIFSGN'2017 – The Sixteenth International Workshop on Intuitionistic Fuzzy Sets and Generalized Nets, 13–15 September 2017, Warsaw, Poland: Volume 2*. Cham: Springer; 2018. P. 165–175. https://doi.org/10.1007/978-3-319-66824-6_15
 15. Ishibuchi H., Nozaki K., Tanaka H. Distributed Representation of Fuzzy Rules and Its Application to Pattern Classification. *Fuzzy Sets and Systems*. 1992;52(1):21–32. [https://doi.org/10.1016/0165-0114\(92\)90032-Y](https://doi.org/10.1016/0165-0114(92)90032-Y)
 16. Holland J.H. Genetic Algorithms. *Scientific American*. 1992;267:66–72. <https://doi.org/10.1038/scientificamerican0792-66>
 17. Alam T., Qamar Sh., Dixit A., Benaida M. Genetic Algorithm: Reviews, Implementations, and Applications. *International Journal of Engineering Pedagogy*. 2020;10(6):57–77. <https://doi.org/10.3991/ijep.v10i6.14567>
 18. Mamdani E.H., Assilian S. An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller. *International Journal of Man-Machine Studies*. 1975;7(1):1–13. [https://doi.org/10.1016/S0020-7373\(75\)80002-2](https://doi.org/10.1016/S0020-7373(75)80002-2)
 19. Шерстнев П.А., Семенкин Е.С. Самоконфигурируемые алгоритмы генетического программирования с адаптацией на основе истории успеха. *Сибирский аэрокосмический журнал*. 2025;26(1):60–70.
Sherstnev P.A., Semenkin E.S. Self-Configuring Genetic Programming Algorithms with Success History-Based Adaptation. *Siberian Aerospace Journal*. 2025;26(1):60–70.
 20. Breiman L., Friedman J., Olshen R.A., Stone Ch.J. *Classification and Regression Trees*. New York: Chapman & Hall/CRC; 1984. 368 p. <https://doi.org/10.1201/9781315139470>
 21. Захарова О.И., Артюшкина Е.С., Холопов С.В. Деревья решений и алгоритмы их построения. *Евразийское научное объединение*. 2020;(4–2):97–99.
 22. Карасева Т.С. Самонастраивающийся алгоритм генетического программирования для решения задачи Коши и вариационной задачи в символьном виде. *Труды Института системного анализа Российской академии наук*. 2019;69(3):80–90. <https://doi.org/10.14357/20790279190307>
Karaseva T.S. Self-Configuring Genetic Programming Algorithm for a Cauchy Problem and Variational Problem in Symbolic Form. *Proceedings of the Institute for Systems Analysis Russian Academy of Sciences*. 2019;69(3):80–90. (In Russ.). <https://doi.org/10.14357/20790279190307>
 23. Mitrofanov S., Semenkin E.S. Decision Tree Pruning Method Using Delayed Sampling. In: *2024 International Conference on Information Technologies (InfoTech), 11–12 September 2024, Sofia, Bulgaria*. IEEE; 2024. P. 1–4. <https://doi.org/10.1109/InfoTech63258.2024.10701393>
 24. Storn R., Price K. Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization*. 1997;11(4):341–359. <https://doi.org/10.1023/A:1008202821328>

25. Sherstnev P.A. Self-Configuring Evolutionary Algorithms Based Design of Hybrid Interpretable Machine Learning Models. In: *Hybrid Methods of Modeling and Optimization in Complex Systems: Proceedings of International Workshop "Hybrid Methods of Modeling and Optimization in Complex Systems" (in the Framework of the Eleventh International Conference on Mathematical Models and Their Applications), 22–24 November 2022, Krasnoyarsk, Russia*. European Publisher; 2023. P. 313–321. <https://doi.org/10.15405/epct.23021.38>
26. Tanabe R., Fukunaga A. Success-History Based Parameter Adaptation for Differential Evolution. In: *2013 IEEE Congress on Evolutionary Computation, 20–23 June 2013, Cancun, Mexico*. IEEE; 2013. P. 71–78. <https://doi.org/10.1109/CEC.2013.6557555>
27. Mann H.B., Whitney D.R. On a Test of Whether One of Two Random Variables Is Stochastically Larger than the Other. *Annals of Mathematical Statistics*. 1947;18(1):50–60. <https://doi.org/10.1214/aoms/1177730491>

ИНФОРМАЦИЯ ОБ АВТОРАХ / INFORMATION ABOUT THE AUTHORS

Шерстнев Павел Александрович, инженер-исследователь, Сибирский государственный университет науки и технологий имени академика М.Ф. Решетнева, Красноярск, Российская Федерация.
e-mail: sherstpasha99@gmail.com
ORCID: [0000-0003-2816-9433](https://orcid.org/0000-0003-2816-9433)

Pavel A. Sherstnev, research engineer, Reshetnev Siberian State University of Science and Technology, Krasnoyarsk, the Russian Federation.

Семенкин Евгений Станиславович, доктор технических наук, профессор кафедры системного анализа и исследования операций, Сибирский государственный университет науки и технологий имени академика М.Ф. Решетнева, Красноярск, Российская Федерация.
e-mail: eugenesemenkin@yandex.ru
ORCID: [0000-0002-3776-5707](https://orcid.org/0000-0002-3776-5707)

Evgeny S. Semenkin, Doctor of Engineering Sciences, Professor of the Department of Systems Analysis and Operations Research, Reshetnev Siberian State University of Science and Technology, Krasnoyarsk, the Russian Federation.

Митрофанов Сергей Александрович, инженер-исследователь, Сибирский государственный университет науки и технологий имени академика М.Ф. Решетнева, Красноярск, Российская Федерация.
e-mail: sergeimitrofanov95@gmail.com
ORCID: [0000-0002-9420-5714](https://orcid.org/0000-0002-9420-5714)

Sergey A. Mitrofanov, research engineer, Reshetnev Siberian State University of Science and Technology, Krasnoyarsk, the Russian Federation.

Ганчев Тодор Димитров, PhD, профессор, заведующий лабораторией искусственного интеллекта, Технический университет – Варна, Варна, Республика Болгария.
e-mail: tganchev@tu-varna.bg
ORCID: [0000-0003-1049-4801](https://orcid.org/0000-0003-1049-4801)

Todor D. Ganchev, PhD, Professor, Head of the Artificial Intelligence Laboratory, Technical University of Varna, Varna, the Republic of Bulgaria.

Статья поступила в редакцию 06.05.2025; одобрена после рецензирования 23.05.2025; принята к публикации 28.05.2025.

The article was submitted 06.05.2025; approved after reviewing 23.05.2025; accepted for publication 28.05.2025.