

## СОДЕРЖАНИЕ

СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ.....	2
ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ .....	3
ВВЕДЕНИЕ .....	5
1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ПОСТАНОВКА ЗАДАЧИ .....	7
1.1 Задача исследования.....	7
1.2 Основные понятия исследования .....	7
1.3 Нормативные документы .....	10
1.4 Определение актуальности и проблемы исследования .....	10
1.5 Научные работы и примеры аналогичных исследований.....	12
2 МЕТОДОЛОГИЯ ИССЛЕДОВАНИЯ ПРЕДМЕТНОЙ ОБЛАСТИ И МЕТОД.....	15
2.1 Структура методологии .....	15
2.2 Обзор существующих методов.....	16
2.3 Метод размещения точек притяжения.....	21
2.3.1 Описание общих требований к входным данным.....	23
2.3.2 Заданный допуск .....	25
2.3.3 Описание алгоритма метода.....	26
3 Реализация метода размещения точек притяжения.....	32
3.1 Рекомендации к внедрению разработанного метода .....	46
ЗАКЛЮЧЕНИЕ.....	50

## **СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ**

ООН – Организация объединенных наций

УДС – Улично-дорожная сеть

ГИС – Геопространственные информационные системы

ДДУ – Детские дошкольные учреждения

МО – Машинное обучение

DBSCAN – Density-based spatial clustering of applications with noise (Алгоритм пространственной кластеризации по шуму)

КРТ – Комплексное развитие территории

РТ – Развитие территорий

## ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

*Аттрактор* – в контексте данной работы – предсказываемые точки притяжения

*Валидация* – подтверждение на основе представления объективных свидетельств того, что требования, предназначенные для конкретного применения или использования, выполнены [39]

*Гиперпараметры* – в машинном обучении – параметры, которые задаются до начала обучения модели и не изменяются в процессе обучения. Они управляют общей структурой и поведением алгоритма, в отличие от параметров модели, которые настраиваются на обучающих данных [41]

*Граф* – структура данных, состоящая из двух компонентов: вершин и ребер [20]

*Датафрейм (DataFrame)* – структура данных, представленная в виде таблицы с строками и столбцами

*Датасет* – набор данных, который используется в различных видах анализа и машинного обучения

*Заданный допуск* – в контексте данной работы – буфер определённого размера, построенный на этапе подготовки данных для рассматриваемых объектов (точек)

*Классификация* – статистический метод, используемый для определения категории новых наблюдений на основе обучающих данных

*Морфотип застройки* — концепция, описывающая типологические и морфологические характеристики городской застройки. Она включает в себя такие аспекты, как плотность и высотность зданий, их размещение относительно друг друга, структуру и конфигурацию улиц, размеры и формы кварталов, а также использование пространства (жилая, коммерческая, общественная и т.д.)

*Нейронная сеть* – сложная дифференцируемая функция, задающая отображение из исходного признакового пространства в пространство ответов, все параметры которой могут настраиваться одновременно и взаимосвязанно [20]

*Тег* – в контексте данной работы – типы объектов, оказывающих влияние на размещение точек притяжения

*Устойчивое развитие* – обеспечение безопасности и комфортных условий проживания человека в процессе градостроительной деятельности, а также минимизация отрицательного воздействия экономической и других видов деятельности на окружающую среду [1]

*Supervised machine learning* – (контролируемое машинное обучение) — это метод машинного обучения, при котором алгоритм обучается на размеченных данных (labeled

data), то есть на входных примерах, для которых известны правильные выходные значения (целевые метки) [34]

## ВВЕДЕНИЕ

Современные города и территории, находящиеся в стадии активного развития, часто сталкиваются с проблемой недостаточного распределения точек притяжения при наличии базовой инфраструктуры – дорог, инженерных сетей, жилых и социальных объектов. Такая ситуация характерна для новых районов, территорий комплексного развития, а также зон, включённых в мастер-планы, но ещё не освоенных в полной мере [53, 54].

Существующие методы анализа данных и моделирования городских пространств оказываются малоэффективными для слабоурбанизированных территорий из-за нехватки данных, нормативные документы и методические рекомендации подходят для планирования таких территорий, но их использование сопряжено с существенной степенью неопределенности. В результате возникают дисбалансы: одни районы перенасыщены коммерческими и общественными объектами, а другие испытывают их острый дефицит, что снижает качество городской среды, а также инвестиционную привлекательность территорий [15].

Современные методы анализа данных и моделирования городских пространств в основном ориентированы на урбанизированные территории, где уже существует развитая инфраструктура и доступ к большому объёму данных [19]. Однако в условиях активного градостроительного развития и девелоперской деятельности значимость приобретают методы, позволяющие определять оптимальное размещение точек притяжения на территориях, где они пока отсутствуют, но запланированы к развитию. Планирование размещения точек притяжения на таких территориях, требует особые подходы и методы, которые должны быть гибкими, адаптивными и способными работать с ограниченным количеством данных.

Машинное обучение, как одна из самых динамично развивающихся областей, демонстрирует высокую эффективность в выполнении различных задач, связанных с анализом данных. В частности, нейросети могут быть использованы для моделирования и оптимизации процессов, даже при ограниченных данных, подстраиваясь под заданные условия [20]. Использование МО позволяет создавать адаптивные модели, которые способны извлекать скрытые закономерности в данных, делать прогнозы и оптимизировать процессы планирования, принимая во внимание различные параметры, такие как плотность населения, площадь застройки, наличие социальной инфраструктуры. Особую важность приобретают подходы, сочетающие нормативные градостроительные принципы с методами машинного обучения и пространственного анализа.

Основной гипотезой исследования является утверждение, что разработанный метод трехуровневой генерации точек притяжения, основанный на комбинации графовых

нейронных сетей и supervised machine learning, позволяет повысить точность планирования их размещения на территориях с отсутствующей застройкой по сравнению с традиционными нормативными подходами. А также то, что использование графовых нейронных сетей для анализа пространственных взаимосвязей позволяет выявлять неочевидные закономерности размещения точек притяжения, не учитываемые в нормативных методах.

Цель исследования является разработка метода пространственного размещения точек притяжения с применением машинного обучения при планировании развития городских территорий.

В рамках достижения цели были поставлены следующие задачи:

- Комплексный системный анализ предметной области, включающий анализ основного понятия в научной среде, а также существующих подходов к планированию размещения точек притяжения в градостроительстве;
- Разработка модели генерации точек притяжения, включающая описание исходных данных, требований к ним и процесса их обработки, а также описание алгоритма работы метода с указанием ключевых характеристик;
- Обоснование применения графовых нейронных сетей и supervised machine learning для оптимизации пространственного распределения сервисов;
- Апробация метода на выбранной территории с последующей интерпретацией результатов;
- Разработка рекомендаций ко внедрению метода в градостроительные практики, а также потенциальные возможности использования.

Объектом данного исследования выступают точки притяжения жителей. Предмет исследования – метод размещения точек притяжения.

Научная новизна разработанного метода заключается в том, что метода использует итеративный подход – разработана трехуровневая система (макро-, мезо-, микромасштаб), обеспечивающая переход от стратегического зонирования к точечному размещению объектов путём применения алгоритмов машинного обучения, анализирующих пространственные взаимосвязи объектов.

# **1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ПОСТАНОВКА ЗАДАЧИ**

## **1.1 Задача исследования**

Развитие и рост городов, увеличение доли городского населения, современные вызовы и меняющиеся потребности жителей определили эпоху нашей современности как эпоху урбанистического развития. Стремительность процесса урбанизации определяет задачу эффективного использования городского пространства – оптимального распределения городских объектов и мест притяжения жителей для обеспечения устойчивого развития. Эффективные методы и стратегии пространственного размещения точек притяжения играют ключевую роль в устойчивом развитии урбанизированных территорий, учитывая комплекс различных факторов. В Повестке дня в области устойчивого развития на период до 2030 года семидесятой сессии ООН от 25.09.2015 года указаны цели в области устойчивого развития [1]. 11 цель – Обеспечение открытости, безопасности, жизнестойкости и экологической устойчивости городов и населённых пунктов, где утверждаются определённые задачи и индикаторы. Среди них непосредственный интерес представляют 11.7 и 11.b [2]. Задача 11.7 подразумевает обеспечение доступности, безопасности и открытости зелёных зон и общественных мест для всех пользователей, 11. b – увеличение числа городов и населённых пунктов, осуществляющих планы и стратегии комплексного развития, устранение социальных барьеров, а также повышение эффективности ресурсопользования [2].

Определение оптимальных параметров масштабирования и пространственного распределения точек притяжения, а также анализ их взаимосвязей и совокупного влияния на городскую структуру позволяют исследовать и оценить воздействие морфологии городского пространства на его устойчивость и динамику развития. Трансформация потребностей городского населения, обусловленная различными моделями урбанизационных процессов, детерминирует адаптацию подходов к локализации точек притяжения в зависимости от типа поселения – от агломераций до малых городов [3].

## **1.2 Основные понятия исследования**

В настоящее время в научной литературе отсутствует единое определение понятия «точка притяжения». В наиболее общем смысле под ней понимается пространственный объект, характеризующийся повышенной посещаемостью благодаря наличию уникальных атрибутов [4]. Анализ существующих исследований демонстрирует терминологическую вариативность: различные авторы включают в данную категорию объекты с общественными, социокультурными, туристическими или коммерческими функциями, а также их комбинации.

В научной литературе наблюдается множественность интерпретаций понятия «точки притяжения». Отечественные исследователи предлагают различные трактовки: Суслов С.И. и Гаврикова К.В. определяют их как качественные публичные пространства с высокой посещаемостью [5]; Громенко И.В. и соавторы акцентируют архитектурно-ландшафтную значимость городских площадей [6]; Тяглов С.Г. и коллеги выделяют социально-активные общественные пространства с развитой инфраструктурой [7]; Амосова Е.В. рассматривает потенциал ревитализации промышленных зон [8]; Лаврусевич П.Е. и Скалабан И.А. анализируют рекреационные лесные массивы [9].

В зарубежных исследованиях встречаются термины "hotspots", "POI" (Points of interest), "attractors". Cai L. и соавторы связывают точки притяжения с зонами экономической активности [10], Alhazzani M. и коллеги предлагают типологию по масштабу влияния [11]. Существующие различия в трактовках свидетельствуют о сложной, многогранной природе данного понятия в градостроительной науке.

При системном анализе предметной области [11, 12, 13] было выявлено несколько вариантов классификации точек притяжения: иерархическая (Рисунок 1), коммуникативно-пространственная (Рисунок 2) и функциональная (Рисунок 3).

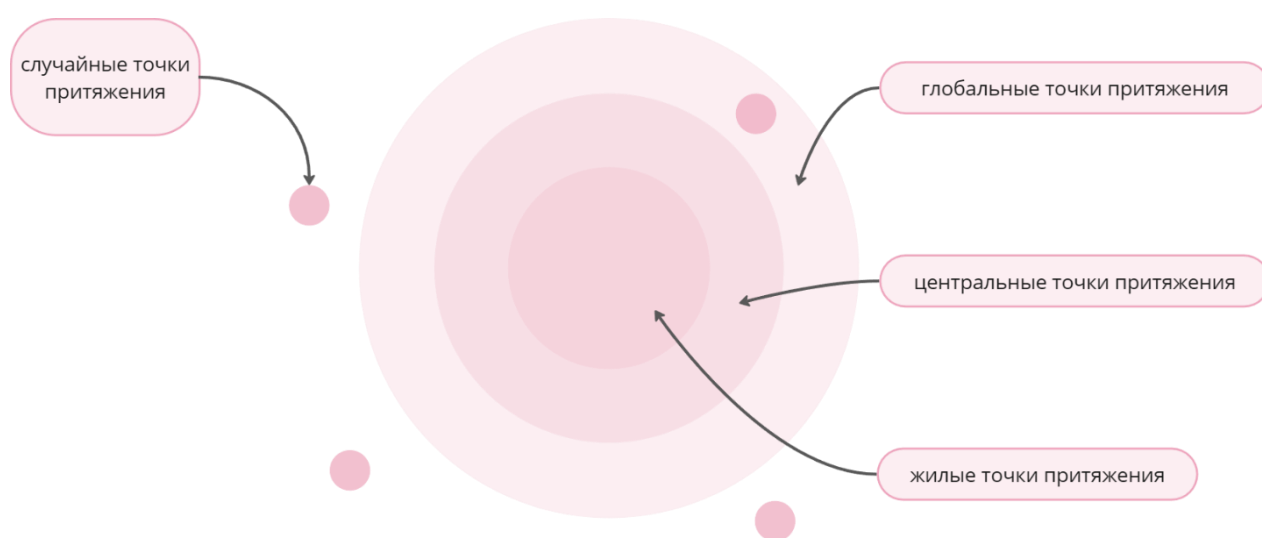


Рисунок 1 – Иерархическая классификация точек притяжения



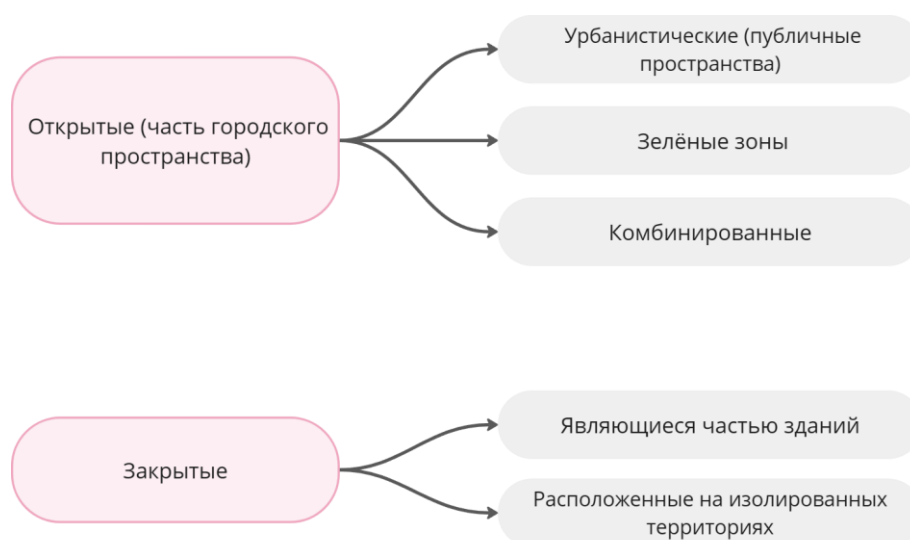


Рисунок 2 – Коммуникативно-пространственная классификация точек притяжения

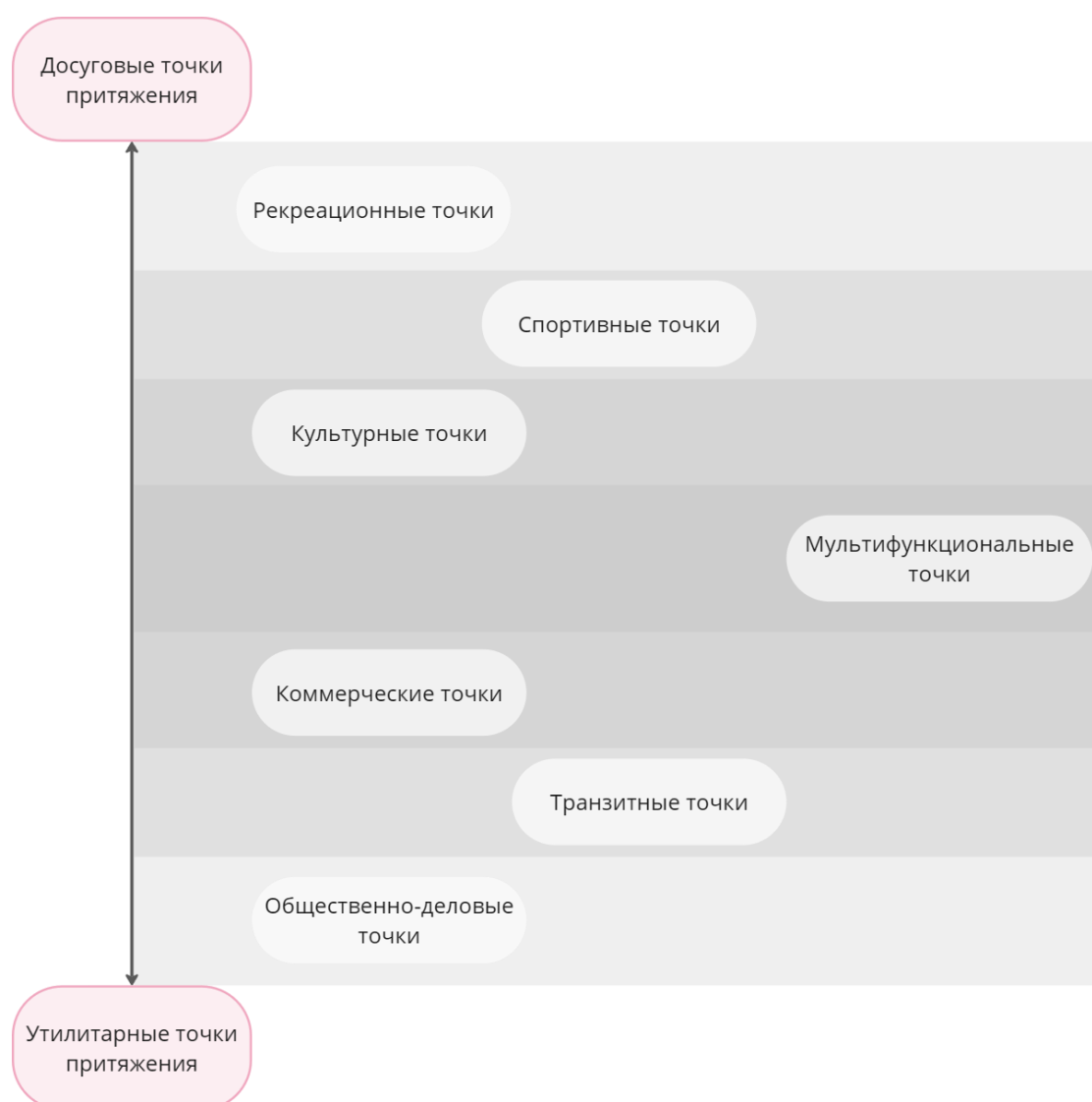


Рисунок 3 – Функциональная классификация точек притяжения

### **1.3 Нормативные документы**

При разработке стратегий пространственного развития территорий, включая как зоны сложившейся застройки, так и неосвоенные участки, применяется нормативно-правовая база, включающая Градостроительный кодекс Российской Федерации, законодательные акты в сфере градостроительной деятельности, а также методические рекомендации, в частности, Стандарт комплексного развития территорий [14]. Указанные документы формируют юридическую основу территориального планирования, устанавливая регламентные требования и нормативные параметры, обязательные для соблюдения при проектировании.

В исследовании проанализированы положения Градостроительного кодекса РФ, регулирующие развитие общественно-деловых и рекреационных зон как ключевых элементов повышения качества городской среды (ст. 29.2, 35 ГрК РФ). Особое внимание кодекс уделяет нормативам землеотвода под социально-культурные объекты (образовательные, медицинские, спортивные учреждения, зелёные зоны). В условиях сложившейся застройки особое внимание уделяется реконструктивным мероприятиям (ст. 40.1, Гл. 6 ГрК РФ), включая модернизацию инфраструктуры с обязательным учётом принципов пешеходной доступности (ст. 10, 23 ГрК РФ) для создания безопасных и комфортных публичных пространств.

Важным документом является СП 42.13330 «Градостроительство. Планировка и застройка городских и сельских поселений» [17], в частности положения, регламентирующие организацию общественно-деловых зон (п.6-7). Согласно нормативу, данные зоны предназначены для размещения:

1. социально-значимых объектов (здравоохранение, образование, культура);
2. объектов обслуживания населения (торговля, общественное питание, бытовые услуги);
3. административных и деловых функций;
4. научно-исследовательских учреждений.

В исторических городах (Санкт-Петербург, Москва, Ярославль и др.) в состав таких зон могут интегрироваться объекты культурного наследия.

### **1.4 Определение актуальности и проблемы исследования**

С 2020 года в градостроительной практике Российской Федерации реализуется механизм комплексного развития территорий, законодательно закреплённый Федеральным законом от 30.12.2020 № 494-ФЗ [16]. Данный подход применяется как в условиях сложившейся застройки, так и при освоении новых территорий [17].

Процесс планирования городского развития требует учёта взаимосвязанных факторов: экономико-географических, социальных, производственных, историко-архитектурных и природно-ресурсных [17]. Такой комплексный анализ обуславливает необходимость системного подхода к функциональному зонированию, включая размещение точек притяжения населения.

Научно доказано, что качество городской среды оказывает непосредственное влияние на психофизическое состояние жителей [18]. В этой связи оптимизация размещения точек притяжения в рамках территориального планирования представляет собой ключевой инструмент формирования комфортной и многофункциональной городской среды.

Процесс размещения точек притяжения при планировании развития урбанизированных территорий происходит путём использования различных методов анализа территории, а также с учётом градостроительных законов. Однако современные методы анализа данных и пространственного моделирования, а также законы и методические указания преимущественно разработаны для урбанизированных территорий с развитой инфраструктурой и значительными массивами доступных данных [19]. Их применение к слабоурбанизированным территориям сталкивается с ограничениями, связанными с частичным или полным отсутствием релевантных данных.

Это обуславливает необходимость разработки специализированных подходов к планированию точек притяжения на таких территориях, характеризующихся гибкостью, адаптивностью и способностью функционировать в условиях дефицита информации.

В данном контексте перспективным направлением представляется применение методов машинного обучения, демонстрирующих высокую эффективность при решении задач пространственного анализа [20]. Нейросетевые модели обладают значительным потенциалом для:

- выявления скрытых закономерностей в условиях ограниченных данных;
- прогнозирования параметров развития территорий;
- оптимизации планировочных решений с учётом множества факторов (плотность населения, характеристики застройки, социальная инфраструктура).

При этом существующие нормативные документы и методические рекомендации, хотя и применимы к планированию слабоурбанизированных территорий, не обеспечивают достаточной точности из-за высокой степени неопределённости, связанной с отсутствием достоверных исходных данных. Это подчеркивает необходимость разработки новых адаптивных инструментов пространственного анализа, способных эффективно работать в условиях неполноты данных.

## 1.5 Научные работы и примеры аналогичных исследований

Ряд изученных исследований описываемой предметной области отлично иллюстрируют подход к кластеризации точек притяжения, а также поиск корреляций с другими городскими объектами. Значимым аспектом анализа научных публикаций явилось изучение потенциала использования машинного обучения в области градостроительного планирования.

Стоит отметить исследование “GSEN: An Ensemble Deep Learning Benchmark Model for Urban Hotspots Spatiotemporal Prediction” [21] (GSEN: ансамблевая эталонная модель глубокого обучения для пространственно-временного прогнозирования городских «горячих точек») авторства Jin G. и коллег, которые разработали ансамблевую модель глубокого обучения GSEN (Geographic-Semantic Ensemble Neural Network) для прогнозирования динамики городских точек притяжения [21]. Модель интегрирует три ключевых компонента: PredRNN для временных зависимостей, GC-PredRNN для пространственных связей и ансамблевый слой для комплексного анализа. Такой подход позволяет одновременно учитывать географические закономерности и семантические паттерны в данных. Эксперименты подтвердили эффективность GSEN в задачах пространственно-временного прогнозирования, демонстрируя ее устойчивость к различным городским контекстам. Модель показывает перспективность комбинирования разных архитектур нейросетей для точного предсказания возникновения и динамики точек притяжения.

Не меньшего внимания заслуживает работа Гриневой Н.В. и соавторов, где ведётся исследование интерпретируемости моделей машинного обучения для задач геоаналитики: моделирования размещения торговых точек розничной сети [22]. В статье рассматривается проблема выбора оптимальных мест для размещения новых торговых точек розничных сетей. Авторы подчёркивают, что традиционные методы (экспертные оценки, статистический анализ) часто недостаточно точны и не учитывают комплекс факторов. Машинное обучение позволяет улучшить прогнозирование, но «чёрный ящик» моделей затрудняет их практическое применение. Поэтому ключевая задача исследования – разработка интерпретируемой модели МО, сочетающей высокую точность с прозрачностью решений. В качестве методологической базы исследования выступает синтез методов геоаналитики и современных алгоритмов МО. Особое внимание уделяется проблеме интерпретируемости моделей, что реализуется через применение методов post-hoc анализа, в частности SHAP (SHapley Additive exPlanations) и LIME (Local Interpretable Model-agnostic Explanations). Для построения прогностической модели проводится сравнительный анализ эффективности различных алгоритмов МО, с особым акцентом на

методы ансамблевого обучения, а именно градиентный бустинг (XGBoost, CatBoost), демонстрирующие наилучшие результаты при работе с табличными данными, содержащими пространственные характеристики. Экспериментальная часть исследования демонстрирует высокую прогностическую способность разработанной модели, что подтверждается значениями метрик качества (RMS,  $R^2$ ). Применение методов интерпретации позволило выявить нелинейные зависимости между факторами размещения и экономической эффективностью торговых точек. Интересным результатом стало выявление территориальной специфики влияния факторов: в центральных районах городов критическое значение приобретает параметр проходимости, тогда как в спальных районах наибольшую значимость имеют демографические показатели.

Научный интерес представляет работа Синициной И.А. «Адаптация инструментов пространственного синтаксиса для анализа освоения территории» [3], где аналитическое проектирование представлено как инструмент пространственного синтеза, позволяющий исследовать функциональные взаимосвязи в городской структуре на примере Волгограда. Методология предполагает интеграцию статичных элементов (здания) и динамических агентов (жители) в единую модель территориального развития. Ключевым аспектом анализа выступает концепция центральности, оцениваемая через:

- доступность элементов городской среды;
- транзитность улично-дорожной сети (УДС);
- показатели притягательности пространств.

Исследование проводилось на двух масштабных уровнях (районный – 800 м, городской – 5000 м) с использованием геоинформационной системы QGIS и плагина SpaceSyntaxTool.

Выявленные закономерности свидетельствуют о полицентричной структуре Волгограда, где коммерческая активность тяготеет к наиболее доступным сегментам транспортной сети. Полученные данные позволяют прогнозировать векторы территориального развития и оптимизировать размещение общественных функций с учётом пространственного поведения горожан.

Перспективным направлением для совершенствования разрабатываемого метода представляется подход прогнозирования потенциальной посещаемости городских территорий, описанный в статье “Modeling Visit Potential to Predict Hotspots of a Future District” (Моделирование потенциала посещений для прогнозирования «горячих точек» будущего района) [23]. Данная методика позволяет оценивать привлекательность общественных пространств для населения, что является ключевым аспектом городской жизнеспособности [24], понимаемой как способность городской среды поддерживать

социальные функции и биологические потребности жителей. Метод представляет собой адаптацию гравитационной модели доступности [25], интегрирующую:

- анализ мультимодальной транспортной подвижности населения;
- оценку пространственного распределения объектов-аттракторов;
- моделирование пешеходных потоков.

Транспортная модель построена на графовом представлении городского пространства, где узлами выступают три категории объектов:

- жилые и рабочие зоны (источники населения);
- объекты притяжения (торговые, рекреационные, гастрономические);
- транспортные узлы (места пересадки).

Применение метода продемонстрировано на примере района LaVallée (Франция) с использованием данных транспортных обследований EGT2010 и платформы мультиагентного моделирования MATSim. Были реализованы три подхода к оценке потенциальной посещаемости:

- по близости к жилым зонам
- по доступности аттракторов
- по интенсивности пешеходных потоков

Наиболее релевантной признана комплексная модель, учитывающая как пространственную доступность, так и параметры пешеходной активности. Полученные результаты позволяют классифицировать общественные пространства по уровню потенциальной посещаемости, что особенно ценно для планирования новых районов, где отсутствуют данные о реальном поведении населения.

## 2 МЕТОДОЛОГИЯ ИССЛЕДОВАНИЯ ПРЕДМЕТНОЙ ОБЛАСТИ И МЕТОД

### 2.1 Структура методологии

На рисунке 4 представлена систематизация существующих методологических подходов к решению задачи размещения точек притяжения. Эти методы можно разделить на две основные группы: нецифровые (традиционные) и цифровые (технологически-ориентированные). К первой категории относятся методы, основанные на субъективных экспертных оценках и социологических исследованиях, обладающие существенными ограничениями в части репрезентативности и объективности получаемых результатов. Вторая группа объединяет современные цифровые подходы, базирующиеся на ГИС и методах data science, что обеспечивает более высокую точность и воспроизводимость анализа.

Разработанный в рамках исследования метод, занимает своё место в предложенной методологии. Его методологическая основа заключается в применении алгоритмов машинного обучения для выявления скрытых паттернов и взаимосвязей в пространственных данных, которые могут быть не очевидны при использовании традиционных аналитических подходов. Новый метод относится к этапу генерации рекомендаций по пространственной организации точек притяжения.

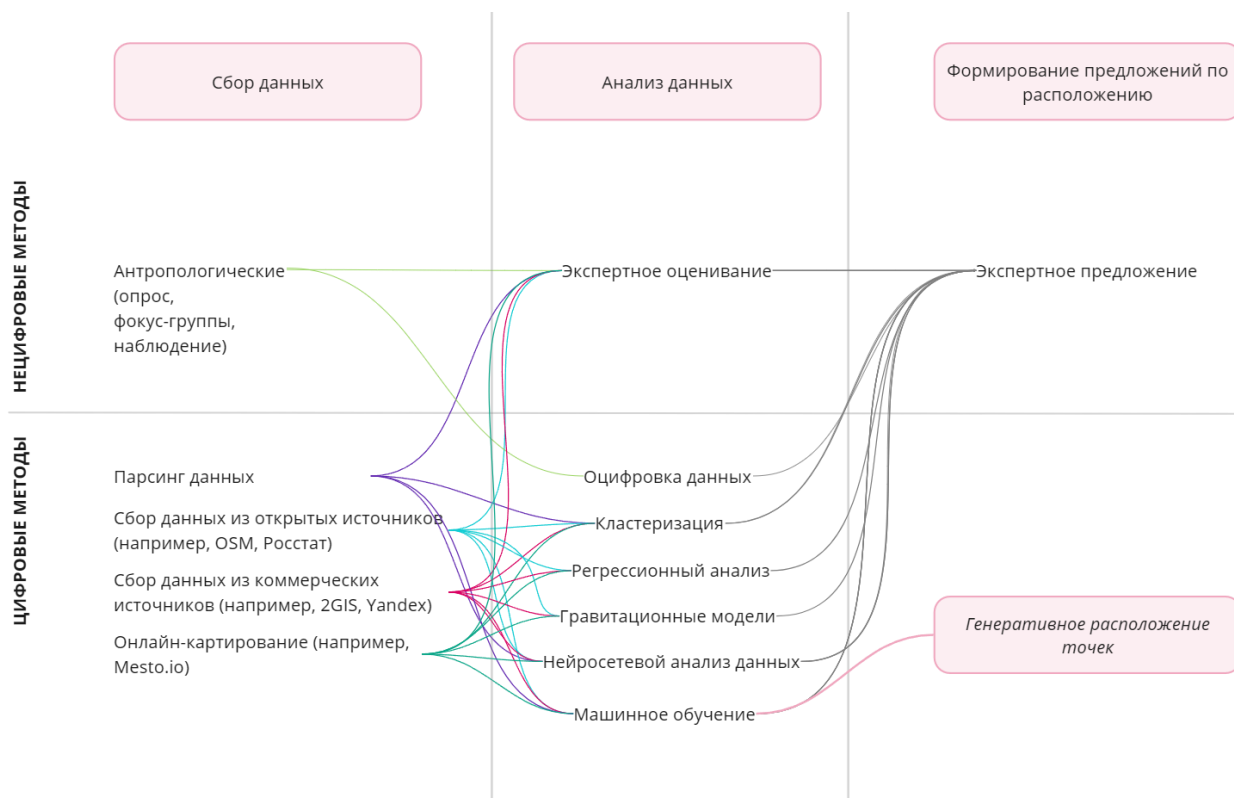


Рисунок 4 – Систематизация существующих методологических подходов

## **2.2 Обзор существующих методов**

Задача размещения точек притяжения проходит в несколько больших этапов:

1. Сбор необходимых данных: на этом этапе реализуется процесс комплексного сбора и систематизации разнородных пространственных данных, формирующих информационную базу исследования. Это могут быть данные о плотности населения, пространственные данные с координатной привязкой объектов, социально-экономические показатели и многое другое (Таблица 1);

2. Анализ собранных данных: этот этап предполагает применение комплексных методов выявления пространственно-временных закономерностей и устойчивых трендов в агрегированных данных (Таблица 2);

3. Формирование предложений по расположению: на основании проведенного комплексного анализа данных разрабатываются рекомендации по оптимальному размещению точек притяжения. Процесс формирования предложений основывается на выявленных статистически значимых закономерностях и учитывает комплекс критериев оптимальности (Таблица 3).



Таблица 1 – Существующие методы сбора данных

Метод	Особенности метода	Преимущества	Недостатки
Антропологический метод	Включает в себя опросы, фокус-группы, наблюдение (включённое и невключённое, стандартизированное и нестандартизированное) [26].	Собирают качественные данные для анализа территории, хорошо работают на урбанизированных территориях. Простота реализации.	Не применимы к неурбанизированным территориям.
Парсинг данных	Парсинг данных о точках притяжения возможен из следующих источников: 1. Карты и отзывы, оставленные на них; 2. Социальные сети – геотметки фотографий, «теги», ключевые слова, анализ пользователей территории.	Экономия времени, обработка большого массива данных [27].	Сложность реализации, постоянные ограничения и изменения со стороны API-приложений [27].
Сбор данных из открытых источников (OSM, Росстат)	Выгрузка данных по определённым критериям, например, теги в OSM, наборы данных в Росстате в формате, удобном для работы в геоинформационных системах (GeoJSON, JSON, CSV).	Свободный и бессрочный доступ к данным и неограниченное дальнейшее использование (в рамках закона).	Для OSM: Данные могут быть неактуальны и/или ограничены, т.к. их добавляют энтузиасты. Теги могут не соответствовать реальности.
Сбор данных из коммерческих источников (2GIS, Яндекс)	Покупка доступа, разрешается использовать в коммерческих целях.	Свежие актуальные данные, большое количество разных типов данных.	Большая стоимость покупки.
Онлайн-картирование (Mesto.io)	Цифровой диалог с пользователями территории. Подразумевает процессы сбора данных, их анализ и визуализации. Использует инструменты онлайн-анкетирования. Платформа настраивается под конкретные цели, является по сути единым комплексным агрегатором открытых геоданных.	Вовлечение граждан в процесс, сбор качественных данных. Есть деление на группы данных (проблема, ценность места, идея).	Возможна сложность в интерпретации результатов.

Таблица 2 – Существующие методы анализа данных

Метод	Особенности метода	Преимущества	Недостатки
Экспертное оценивание	<p>Собранные данные анализируются предметным специалистом (экспертом) для дальнейшей работы. Может быть индивидуальным (оценки основаны на мнениях независимых друг от друга экспертов) и коллективным (оценки основаны на коллективном мнении экспертов). Также делится на однокритериальные и многокритериальные. В качестве ещё одного признака используется тип оценки – ординальный и кардинальный [28].</p> <p>Происходит в несколько этапов:</p> <ol style="list-style-type: none"> <li>1. Подготовка экспертизы;</li> <li>2. Экспертный опрос;</li> <li>3. Анализ экспертной информации.</li> </ol>	Гибкость в принятии решений, наличие интуитивного мышления, способность эксперта строить на основе наблюдений разумные заключения, разумный отбор принимаемых решений [28].	Невозможность детального предсказания последствий; наличие факторов, не поддающихся контролю эксперта; наличие нескольких возможных путей решения и необходимость выбора одного из них [28].
Кластеризация	<p>Фактически, это набор методов для группировки данных по определённым критериям в кластеры, что позволяет выявлять максимальные сходства между исследуемыми объектами, а также упрощать их анализ и визуализацию.</p> <p>Делится на три основных типа:</p> <ol style="list-style-type: none"> <li>1. Разделительные (K-means, CLARANS);</li> <li>2. Иерархические: <ul style="list-style-type: none"> <li>– Агломеративные;</li> <li>– Дивизимные;</li> </ul> </li> </ol> <p>Алгоритмы на основе плотности (DBSCAN, OPTICS, HDBSCAN) [29]</p>	У k-means хорошая скорость работы и чёткое определение кластеров. При иерархической кластеризации в начале не нужно указывать количество кластеров. DBSCAN заранее не требуется количество кластеров, выделяет выбросы как шум и легко находит кластеры произвольной формы и размера [29].	Метод k-means не видит выбросы. К минусам иерархических методов относится их массивность – большая длительность вычислений и требование большого объёма памяти. DBSCAN не подойдёт для «скупенного» набора данных [29].

Продолжение таблицы 2

Гравитационные модели	<p>Различные объекты обладают разной силой притяжения для пользователей, поэтому есть прямая зависимость от выбора модели и рассчитываемой силы притяжения. Выделяются основные гравитационные модели:</p> <ol style="list-style-type: none"> <li>1. Модель Рейли;</li> <li>2. Модель критической точки Конверса;</li> <li>3. Обратная модель критической точки Мейера и Мэйсона;</li> <li>4. Модель Хаффа [30].</li> </ol> <p>В большинстве современных исследований и на практике (как, например, в сети магазинов «Магнит») используется гравитационная модель Хаффа.</p>	<p>Модель Хаффа успешно используется в предиктивных моделях с использованием гео-данных с высокой степенью применимости из-за основополагающей идеи «гравитации» и вероятностной интерпретации результатов [30].</p>	<p>У модели Хаффа есть ограничения: предполагается, что поток однороден и на него влияет только расстояние, а люди или товары – единственные учитываемые факторы [30].</p>
Регрессионный анализ	<p>Данный анализ – поиск эффективных независимых переменных для объяснения зависимых переменных путём запуска инструмента регрессии. Это необходимо, чтобы определить, какие величины являются эффективными предсказателями. Различают следующие виды регрессии:</p> <ol style="list-style-type: none"> <li>1. Линейная;</li> <li>2. Нелинейная: <ul style="list-style-type: none"> <li>– регрессии, нелинейные относительно включенных в анализ объясняющих переменных, но линейные по оцениваемым параметрам;</li> <li>– регрессии, нелинейные по оцениваемым параметрам [31].</li> </ul> </li> </ol>	<p>Простота вычислительных алгоритмов, наглядность и интерпретируемость результатов (для линейной модели) [31].</p>	<p>Невысокая точность прогноза, субъективный характер выбора вида конкретной зависимости. Невозможность объяснения причинно-следственной связи [31].</p>

Продолжение таблицы 2

Нейросетевой анализ данных	<p>Процесс использования нейронных сетей проходит в несколько этапов:</p> <ol style="list-style-type: none"> <li>1. Построение реляционных баз данных на основе исследуемых показателей;</li> <li>2. Разработка архитектуры и параметров нейросети;</li> <li>3. Процесс обучения нейросети;</li> <li>4. Использование нейросетевой модели для прогнозирования исследуемых признаков [32].</li> </ol>	<p>Нейросетевые модели находят сложные зависимости и прогнозируют временные ряды данных [32]. Нейросети самообучаемы, могут применять не очевидные человеку методы и постоянно улучшать результат своей работы, имеют качественную адаптивность, даже при изменении входных данных, также обладают высокой скоростью работы [33].</p>	<p>Требуется много времени для многоцикловой настройки элементов модели и их связей и построения непосредственно модели. Могут возникнуть проблемы при подготовке выборки для обучения в связи с недостаточным количеством доступных материалов. Нейронная сеть «запоминает» ответы, а не выявляет закономерности. Нельзя получить данные о том, как нейросеть пришла к решению. Также НС не может запоминать навыки, и использовать её в динамически изменяющихся средах не получится [33].</p>
Машинное обучение	<p>Путём обработки данных МО могут найти логические взаимосвязи между данными (поиск паттернов). Выделяются методы МО:</p> <ol style="list-style-type: none"> <li>1. Контролируемое обучение («С учителем»);</li> <li>2. Неконтролируемое обучение («Без учителя»);</li> <li>3. Обучение в действии;</li> <li>4. Полууправляемое машинное обучение [34].</li> </ol>	<p>Позволяют в автоматическом режиме восстанавливать структуры и сложные закономерности в данных, которые не видны или не очевидны человеку. МО может строить предсказания, запоминать новое, воспринимать имеющееся, выбирать лучшее [35].</p>	<p>Нет гарантии, что МО будет работать в каждом случае. Очень высокие требования к качеству входных данных [35].</p>

Таблица 3 – Существующий метод по формированию предложений по расположению точек притяжения

Метод	Особенности метода	Преимущества	Недостатки
Экспертное предложение	<p>Процесс оценивания специалистом (экспертом) данных, прошедших анализ и представленных в некоторой визуальной форме.</p>	<p>Индивидуальный подход к размещению точек притяжения.</p>	<p>Затрата большого количества времени для определения оптимального местоположения, субъективность оценки.</p>

### **2.3 Метод размещения точек притяжения**

Предлагаемый метод предназначен для слабоурбанизированных территорий, характеризующихся наличием зданий, объектов социальной инфраструктуры и элементов УДС, но испытывающих дефицит наполнения точками притяжения. Такие условия типичны для территорий, описываемых в материалах генеральных планов и мастер-планов городов, а также в проектах КРТ.

Метод находит практическое применение на стадии предпроектных исследований:

- при планировании жилых районов девелоперскими компаниями;
- для обоснования размещения коммерческих объектов малым и средним бизнесом.

Ключевые возможности метода:

- возможность интеграции с существующими системами градостроительного планирования;
- адаптивность к различным масштабам проектирования;
- научная обоснованность принимаемых решений.

Основная часть метода реализуется в виде итеративного процесса генерации сервисов через трехуровневую систему генерации, отражающую иерархию пространственного планирования. Каждый уровень соответствует определённой степени детализации и учитывает специфические взаимосвязи между рассматриваемыми объектами городской среды. Предлагаемый метод обладает свойством структурной адаптивности, позволяя варьировать состав исходной выборки анализируемых объектов в зависимости от исследовательских задач. Фундаментальным условием их включения в анализ является наличие верифицированных данных, однозначно определяющих типологическую классификацию сервисов.

#### **1. Верхнеуровневая генерация (макромасштаб)**

На данном этапе (Рисунок 5а) генерируются укрупнённые категории точек притяжения (аттракторов), сервисы, пространственное распределение которых демонстрирует значимую корреляцию с исходными данными. В частности, при их размещении учитывается близость к объектам социальной инфраструктуры и узлам транспортного каркаса.

Генерация осуществляется на основе нормативных градостроительных показателей, демографических прогнозов и данных о существующей застройке. Результатом является схема зонирования территории с распределением основных функций.

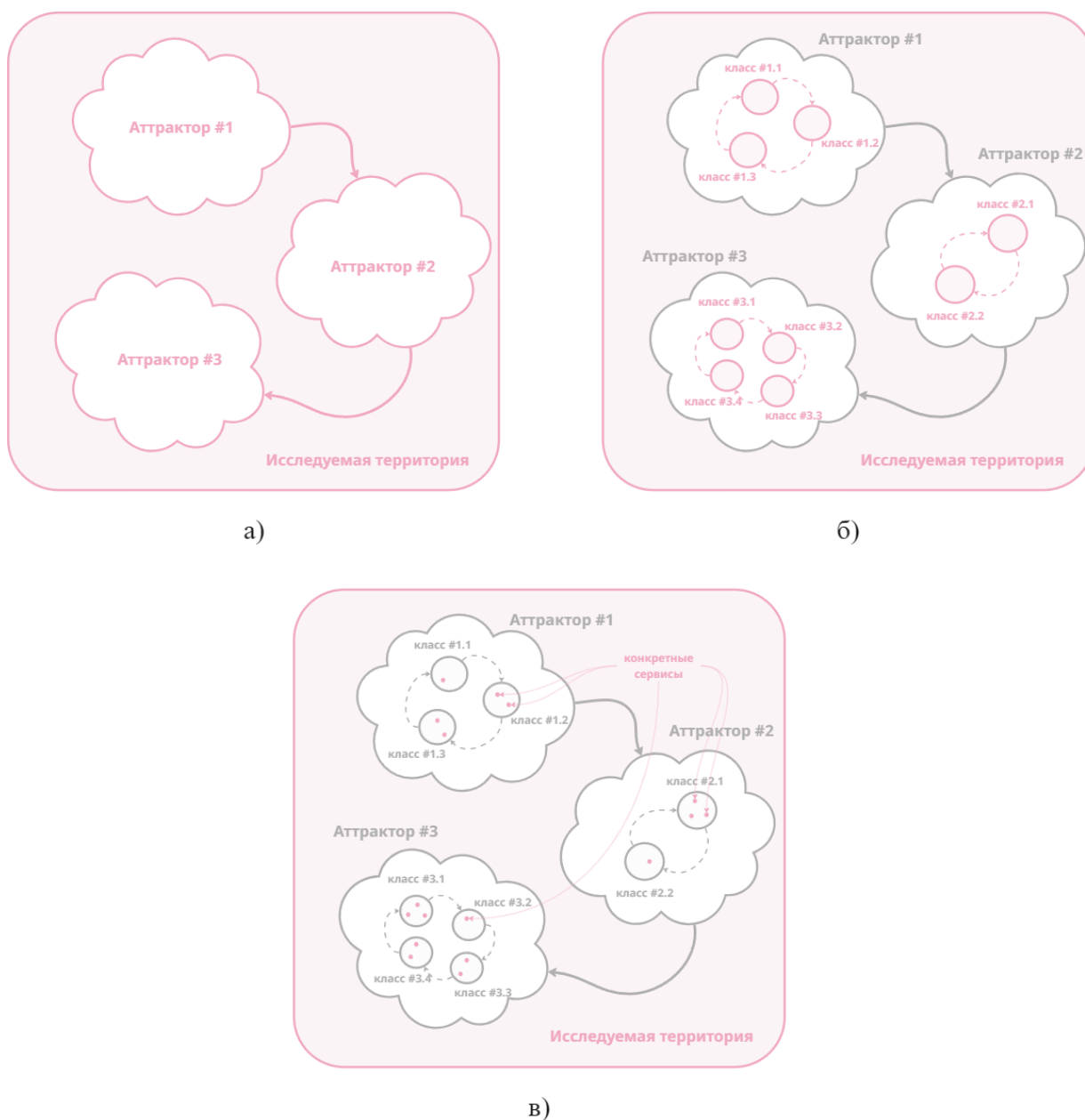


Рисунок 5 – Схематичное представление процесса генерации точек

## 2. Среднеуровневая генерация (мезомасштаб)

На этом уровне (Рисунок 5б) выявляются более детализированные зависимости между сервисами в рамках выделенных функциональных зон. Используются методы пространственного анализа и МО, позволяющие определить оптимальные соотношения между различными типами объектов.

## 3. Точечная генерация (микромасштаб)

Заключительный этап предполагает определение конкретных типов сервисов с учётом выстроенных ранее взаимосвязей на исследуемой территории (Рисунок 5в).

Уровневая структура метода позволяет последовательно переходить от абстрактного планирования к конкретным решениям, минимизируя риски неэффективного использования пространства.

В методе выделено 4 основных этапов: сбор данных по территории, работа с графовой нейронной сетью, работа с Supervised Machine Learning и формирование предложений по пространственному распределению точек притяжения (Рисунок 6). Второй и третий этапы являются описанным выше итеративным процессом генерации.

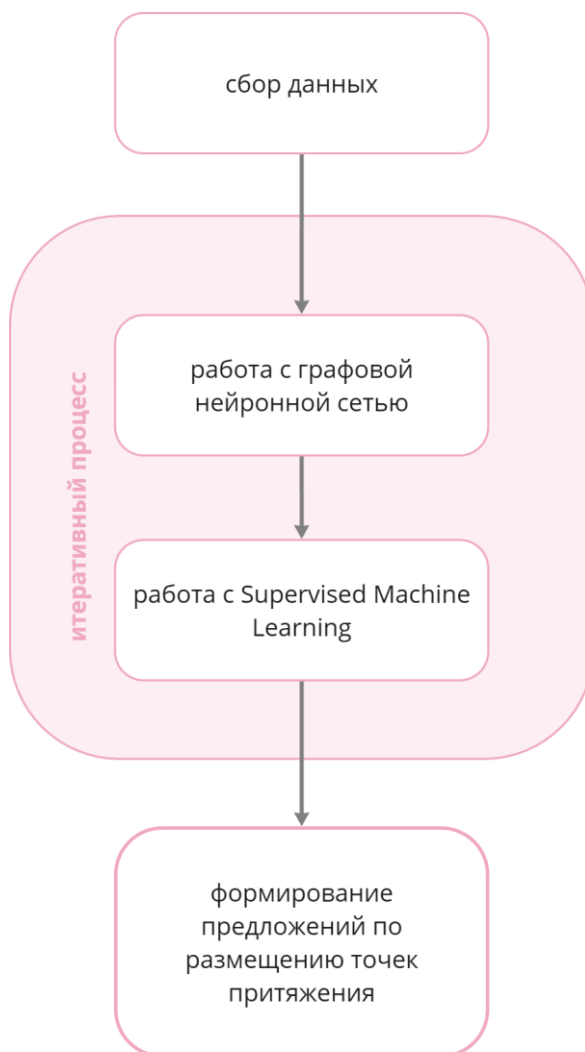


Рисунок 6 – Блок-схема метода

### 2.3.1 Описание общих требований к входным данным

В рамках представленного метода используются исходные данные, зафиксированные в документах территориального планирования – мастер-планах и генеральных планах развития территории. Эти документы содержат ключевые параметры застройки, включая количественные показатели жилого фонда, а также демографические прогнозы, отражённые в технико-экономических показателях [36]. Кроме того, в них регламентированы планировочные решения, такие как конфигурация УДС с указанием транспортных узлов, а также нормативы размещения объектов социальной инфраструктуры [37].

После выявления ключевых характеристик исследуемой территории, осуществляется подготовка данных для обучения модели. Основное условие при

формировании тренировочной выборки — соответствие морфологических параметров застройки данным по исследуемой территории. Информацию по морфотипичности застройки можно получить, используя открытую библиотеку с открытым исходным кодом BlocksNet, реализуемую на языке программирования Python [38]. Библиотека позволяет создать информационную модель города на основе метрик доступности кварталов, а также предоставляет аналитические инструменты для оценки городских сетей, анализа предоставления услуг и оптимизации территориального планирования.

В состав обучающей выборки должны быть включены сведения о жилых массивах, содержащие как демографические показатели, так и пространственные атрибуты с точной геопривязкой. Особое внимание уделяется функциональному наполнению территории — данные о распределении и типах объектов социальной инфраструктуры тех же типов, что в исследуемой территории.

На этом этапе осуществляется сбор пространственных данных о распределении сервисов на тренировочной территории. Включаются следующие параметры:

- точные географические координаты каждого объекта;
- функциональная характеристика (тип сервиса);
- показатели взаимного влияния и корреляционных связей между объектами.

Данное исследование базируется на использовании открытых источников информации, что исключает необходимость работы с конфиденциальными или ограниченными в доступе данными, однако добавляет проблему полноты и точности данных. Основным источником геопро пространственных данных в исследовании выступает платформа OpenStreetMap (OSM). Использование OSM гарантирует воспроизводимость результатов исследования. Открытая архитектура системы предусматривает возможность свободного анализа и обработки пространственных данных.

Полученные из OSM данные подвергаются кластерному анализу, в результате которого формируются укрупнённые функциональные группы. Этот процесс учитывает различные составляющие:

- пространственную близость объектов;
- схожесть функционального назначения.

Сформированные группы становятся базой для последующей уровневой генерации, описанной выше. Особое внимание уделяется сохранению выявленных корреляционных связей при переходе между уровнями генерации.

На этом этапе исследования также осуществляется сбор из OSM пространственных данных о транспортной инфраструктуре. В качестве объектов рассматриваются элементы УДС, включающие:



- остановочные пункты наземного транспорта (автобусные, трамвайные);
- станции метрополитена;
- иные узлы транспортного каркаса территории, предусмотренные характеристиками исследуемой территорией.

На Рисунке 7 представлено схематичное применение метода на новую территорию. Цифрами 1 и 2 отмечена исследуемая территория, а также этап сбора информации о ней. После этого идёт этап применения обученной модели на основе собранных ранее данных. Цифрой 4 отмечен этап итогового формирования предложений по пространственному размещению точек притяжения.

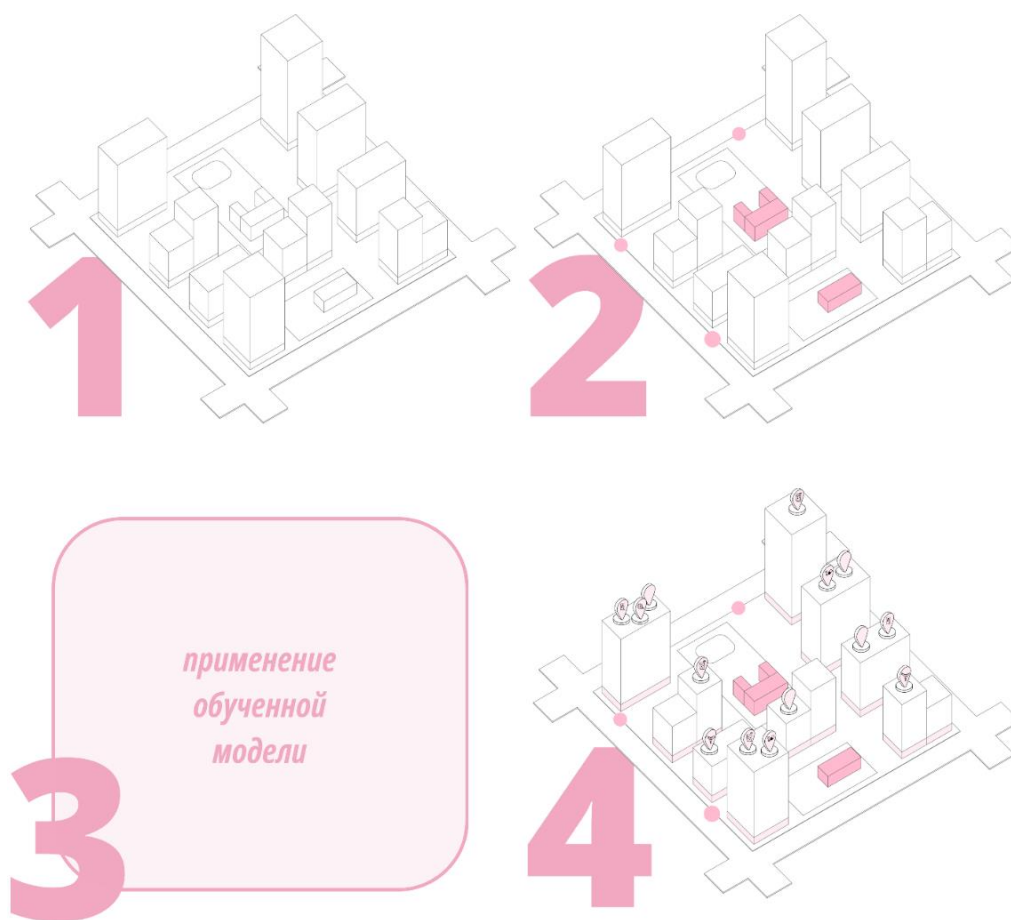


Рисунок 7 – Схематичное отображение применения разработанного метода на новой территории

### 2.3.2 Заданный допуск

Ключевым элементом является определение буфера доступности (соседства) сервисов – заданного допуска. Определение величины данного допуска обусловлено спецификой решаемых исследовательских задач.

Урбанистические исследования показывают, что комфортная пешеходная дистанция до точек притяжения обычно не превышает 5–10 минут ходьбы, что соответствует 400–800

м, однако для локальных сервисов (например, кафе, аптеки) зона спроса сужается до 100–300 м [42–43]. В работах, анализирующих распределение малого бизнеса, буфер 200 м часто используется как оптимальный для прогнозирования размещения локальных сервисов [44].

Согласно теории пространственного синтаксиса, человек эффективно воспринимает объекты в радиусе около 250 м – это пределы чёткой видимости и ориентации в городской среде [45]. В городах с высокой плотностью, например, центральные районы Москвы или Берлина, даже 100 м могут включать десятки зданий, что достаточно для анализа. В пригородах допуск расширяют до 300–500 м из-за разреженной застройки.

Эксперименты с GNN показывают, что для графов, построенных на пространственных данных, радиус связи  $>500$  м приводит к избыточному усреднению признаков, а  $<50$  м – к фрагментации данных [46]. Диапазон 100–300 м балансирует между точностью и вычислительной эффективностью. Доля выбросов растёт при увеличении допуска. Эмпирические тесты подтверждают, что  $200 \pm 100$  м минимизируют ошибки при сохранении репрезентативности [47].

В контексте данного исследования допуски были усреднены до показателя 300 м. Буфер для школ равен 500 метрам, для ДДУ – 300 м [17].

### **2.3.3 Описание алгоритма метода**

Начальный этап исследования предполагает комплексную процедуру сбора и обработки пространственных данных, как было описано в предыдущем разделе. Типы необходимых данных для тренировочных датасетов и их открытые источники представлены на Рисунке 8. Полная схема применения разработанного метода представлена на Рисунке 9.

Ключевые аспекты подготовки данных включают:

1. Систематизацию исходной информации по категориям:
  - параметры застройки – определение морфотипа;
  - транспортная инфраструктура;
  - социально-экономические показатели.
2. Нормализацию данных:
  - унификацию форматов;
  - приведение к единой системе координат;
  - устранение противоречий;
  - удаление дубликатов.

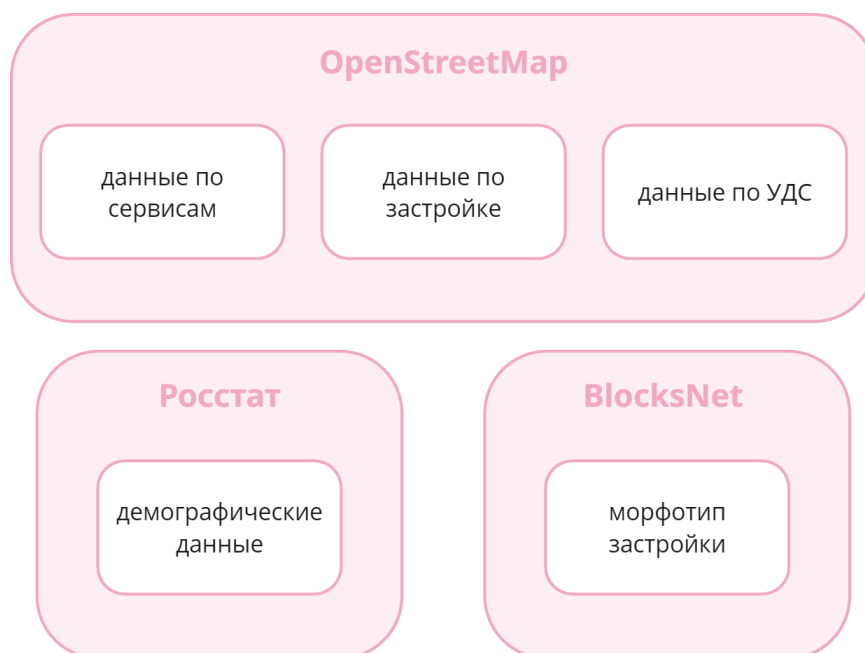


Рисунок 8 – Типы данных и их источники

Процесс обработки пространственных данных включает агрегацию аттракторов в пределы геометрических границ с преобразованием в центроиды. Рассчитывается, какие теги попадают в заданный допуск. Для этого необходимо построить допуски и посчитать количество попавших в них тегов. Транспортные узлы необходимо сохранить отдельно в формате GeoJSON, тогда как остальные данные экспортировать в CSV.

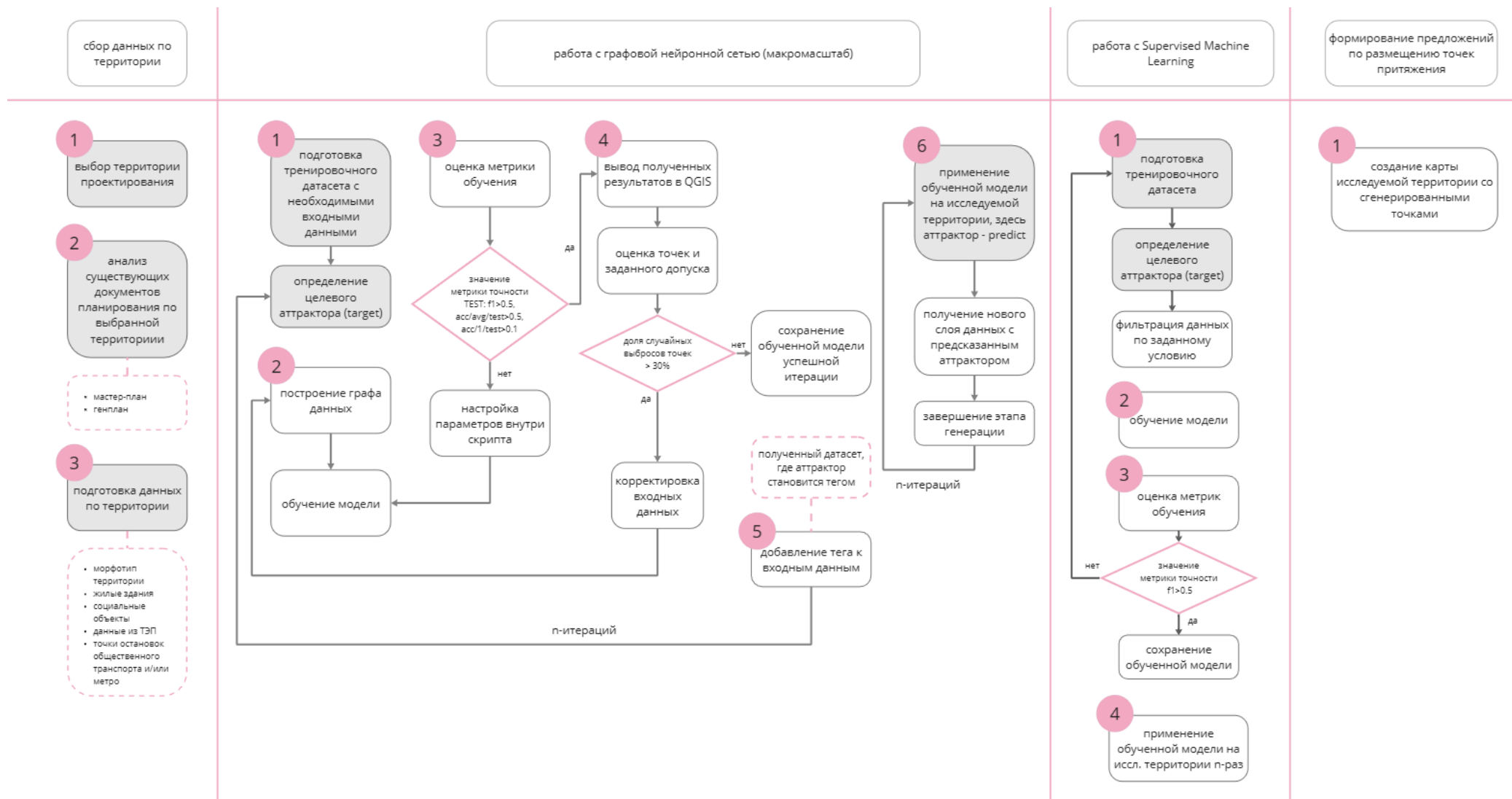


Рисунок 9 – Схема применения разработанного метода

Следующий шаг – работа с графовой нейронной сетью. Каждый итеративный этап представляет собой повтор набора действий. После подготовки тренировочного датасета определяется целевой объект – предсказываемый на данной итерации аттрактор – сервис-тег. Далее применяется двухчастный программный код. Первая часть кода выполняет анализ транспортной инфраструктуры (остановки автобусов и трамваев, станции метро) и близлежащих зданий, в которые была собрана информация о аттракторах и сервисах-тегах (Приложение 1). Код создаёт граф, представляющий взаимодействия между точками зданий и транспортной инфраструктурой. Точки зданий с аттракторами и тегами подаются на граф в качестве узлов, имеющих атрибуты (координаты, площадь жилая и восполненное население). Рёбра соединяют точки, находящиеся в пределах заданного расстояния или ближайшие, если иные поблизости отсутствуют.

Вторая часть кода реализовывает задачу бинарной классификации графа с использованием графовой нейронной сети GCN (Graph Convolutional Network) [20] на библиотеке PyTorch Geometric (Приложение 2). Функция загружает граф из файла, подготовленного на предыдущем шаге работы с кодом, с помощью модуля pickle и возвращает объект графа, представленный библиотекой NetworkX. Создается список атрибутов узлов, таких как площадь, жилая площадь, количество тегов и т.д. Также добавляется тип узла (остановка автобуса, трамвая или метро) и бинарный признак наличия аттрактора. Модель GCN реализует графовую сверточную сеть с n-слоями. Между слоями применяются нормализация и функция активации. Далее происходит обучение модели и оценка, выводимая после каждой эпохи.

После реализации второй части кода происходит оценка метрики обучения – F1-метрика, точность (accuracy) и точность для класса 1. При значениях TEST:  $f1 > 0.5$ ,  $acc/avg/test > 0.5$ ,  $acc/1/test > 0.1$  модель считается обученной. При результатах ниже приведённых необходимо скорректировать параметры внутри скрипта и обучить модель повторно. Если результаты обучения находятся в пределах допустимых они выводятся в QGIS (или иную ГИС-программу) для дальнейшего анализа. Для вывода тренировочных и тестовых выборок используется часть кода, представленная на Рисунке 10.

```
print('saving...')
df_out.iloc[data_.train_mask].to_csv('/content/train.csv', index=False)
df_out.iloc[data_.test_mask].to_csv('/content/test.csv', index=False)
```

Рисунок 10 – Код для вывода тренировочных и тестовых выборок

В программную среду QGIS загружаются тестовая выборка, полученная на предыдущем этапе обработки данных, и исходный набор пространственных данных. С помощью инструментов обработки векторных данных создается буферная зона с параметрами, соответствующими заданному допуску, который использовался при

подготовке исходного датасета. Анализ включает исследование атрибутивной таблицы и визуальную оценку пространственного распределения объектов, что позволяет выявить точечные выбросы. Часть этих аномалий связана с особенностями расположения зданий-аттракторов (predict), которые находятся в непосредственной близости от других зданий со схожими или идентичными характеристиками. Этот эффект обусловлен тем, что заданный допуск охватывает не только целевые здания-аттракторы, но и прилегающие объекты, не обладающие искомыми свойствами, но имеющие сходные атрибутивные признаки.

Если полученное значение выбросов не превышает 30%, модель считается адекватной и успешно обученной. Она сохраняется для дальнейшего применения на данных исследуемой территории. В противном случае требуется дополнительная проверка данных, уточнение параметров модели и дообучение. Долю выбросов можно рассчитать по следующей формуле:

$$\text{Доля выбросов} = \frac{N \times 100\%}{N_{total}}, \quad (1)$$

где  $N_{total}$  – количество всех сгенерированных точек;  $N$  – количество сгенерированных точек, не попавших в заданный допуск.

Для следующей итерации используется датасет, где только что предсказанный аттрактор теперь становится сервисом-тегом, оказывающим на размещение нового predict влияние. Таким образом повторяется n-количество итераций.

После завершения генерации сервисов на макромасштабном уровне выполняется переход к среднему уровню детализации, который включает классификацию сформированных укрупнённых групп. Данный этап реализуется посредством использования программного кода, предназначенного для решения задачи классификации с использованием ансамбля методов машинного обучения, включая Random Forest, XGBoost, LightGBM и CatBoost.

Исходные данные загружаются из CSV-файла в объект DataFrame, после чего выполняется их предварительная обработка, включающая фильтрацию записей по заданному условию и проверку наличия данных после фильтрации. Заданное условие определяется задачей конкретной итерации. На этапе подготовки данных формируются матрицы признаков и целевой переменной, а также осуществляется разделение на обучающую и тестовую выборки. Для каждого из используемых алгоритмов определяется индивидуальная сетка гиперпараметров [41]. Процесс обучения реализован с применением GridSearchCV, что позволяет систематически перебирать комбинации гиперпараметров для нахождения оптимальной конфигурации каждой модели. В качестве целевой метрики выбрана F1, что особенно важно при работе с потенциально несбалансированными классами. Для алгоритма CatBoost предусмотрена специальная процедура настройки,

учитывающая обработку категориальных признаков. В дополнение к стандартному выводу предсказанных классов (бинарных меток 0 или 1) для каждой модели, в коде реализована возможность получения вероятности принадлежности к положительному классу (*prediction\_probability*). Это достигается с помощью метода *.predict\_proba()*, который возвращает матрицу размера (*n\_samples, n\_classes*) с вероятностями для каждого класса. Это позволяет оценивать не только итоговый класс, но и степень уверенности модели в предсказании, что важно для задач с дисбалансом или требующих кастомизации порога классификации. Значения сохраняются в результирующем датафрейме и могут быть использованы для пост-анализа.

После обучения моделей выполняется оценка модели с расчётом ключевых метрик качества (F1-score и accuracy). На заключительном этапе производится сравнение моделей по метрике F1-score и сохранение лучшей из них. Алгоритм итеративного применения обученных моделей на новую территорию аналогичен процессу макроуровня, описанному выше.

### 3 Реализация метода размещения точек притяжения

Для демонстрации практической применимости разработанного метода было отобрано несколько сопредельных жилых комплексов в юго-западной части Санкт-Петербурга, включая существующую застройку ЖК «Солнечный город» и прилегающие жилые массивы («Клёны», «Константиновское», «Английская миля»). Выбор указанных территориальных объектов обусловлен необходимостью эмпирической верификации результатов моделирования в условиях реальной городской застройки. Был проведён сбор данных, включающих морфологические характеристики застройки, демографические показатели, элементы транспортной инфраструктуры и объекты социального обслуживания (Рисунок 11).

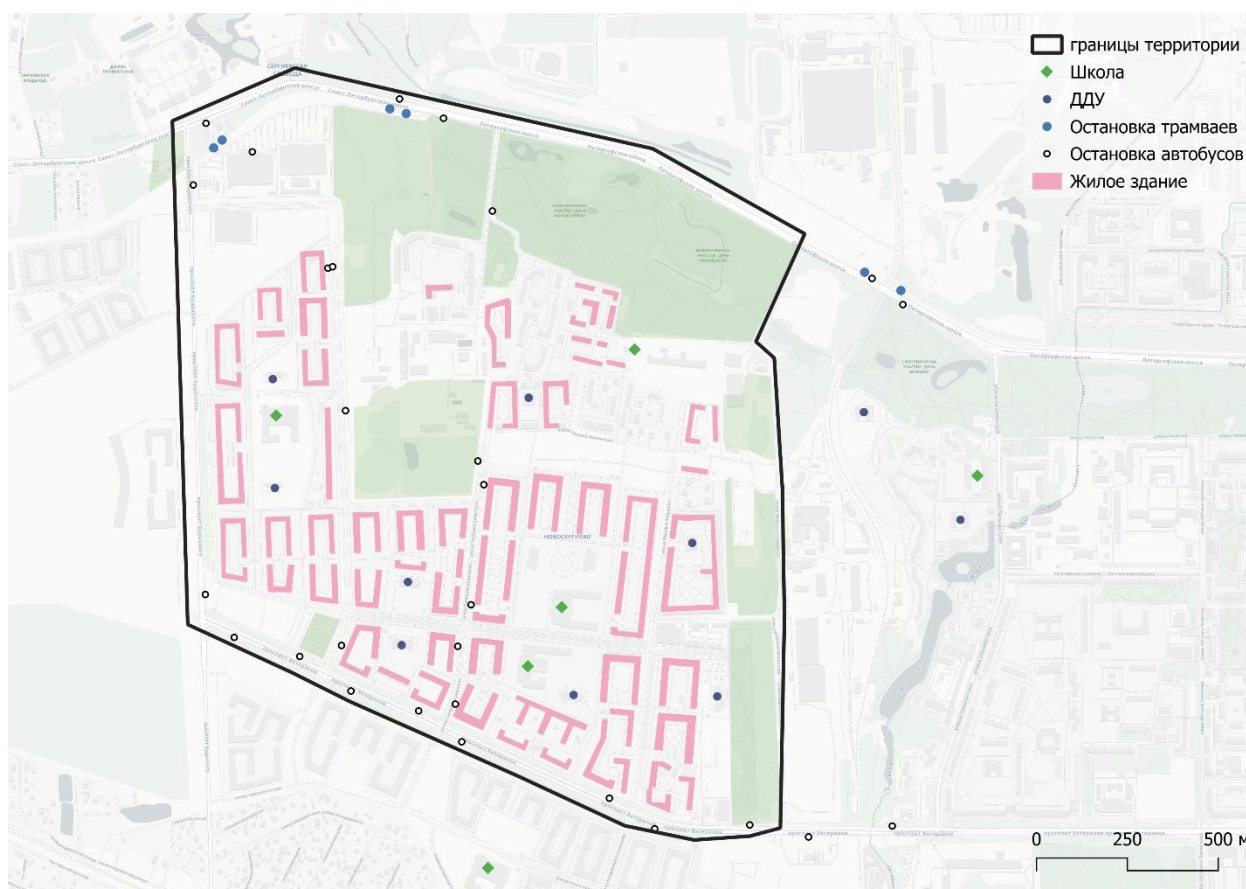


Рисунок 11 – Схема исследуемой территории ЖК «Солнечный город» и части прилегающих жилых комплексов

Критически важным аспектом методологии стало применение модели к специально выделенному тестовому участку, изначально исключенному из тренировочного набора данных. Такой подход обеспечивает два принципиальных преимущества: во-первых, полностью исключается возможность переобучения модели; во-вторых, гарантируется объективность процедуры валидации.

После сбора первичной информации о территории и определении характеристик, необходимых для включения в обучение модели, были сформированы тренировочные



датасеты для верхнего уровня генерации. Для подготовки датасетов были собраны данные из OSM по отобранным заранее тегам. Специфика объектов социальной инфраструктуры выбранной территории определила дальнейшие укрупнённые группы аттракторов:

- учреждения дополнительного образования;
- торгово-коммерческие сервисы;
- объекты здравоохранения;
- сервисы питания;
- финансовые сервисы;
- сервисы досуга и развлечений.

Для обеспечения максимальной полноты исходных данных без потери информации в процессе экспорта, запросы формировались в среде OverpassTurbo с использованием консольного интерфейса. Например, для получения данных о торговых объектах применялся запрос верхнеуровневого тега `shop=*`, позволяющий экспортировать все категории коммерческих точек. Для данных по учреждениям детского и дополнительного образования использовались конкретные теги: `amenity=dancing_school`, `amenity=language_school`, `amenity=library`, `amenity=music_school`, `amenity=training`. По такой методологии выгружались остальные типы указанных выше сервисов.

Важно отметить, что выгрузка включала не только точечные объекты, но и полигональные элементы, используемые в системе OSM для обозначения некоторых объектов. Последующая обработка данных в QGIS предусматривала процедуру пространственного объединения полигонов и точек, что позволило устранить потенциальные дублирования объектов и обеспечить целостность данных.

В рамках исследования также осуществлена выгрузка данных о транспортных узлах улично-дорожной сети и объектах социальной инфраструктуры, включая образовательные учреждения (школы и дошкольные образовательные организации), что обеспечило сопоставимость с тестовой территорией.

Для получения данных о жилом фонде и соответствующих демографических показателях была задействована специализированная платформа «Цифровая урбанистика».

Последующий этап исследования потребовал пространственного ограничения анализируемой территории в соответствии с заданным морфотипом застройки, а именно – многоэтажным жилым строительством. Для реализации этой задачи была выполнена выгрузка матрицы кварталов SpaceMatrix по территории Санкт-Петербурга с использованием библиотеки BlocksNet.

Полученные данные были импортированы в геоинформационную систему QGIS, где средствами атрибутивного анализа осуществлен отбор кварталов, соответствующих

заданному морфотипу. Данная процедура обеспечила точное выделение зоны исследования согласно установленным критериям морфологической классификации, что является необходимым условием для последующего этапа работы.

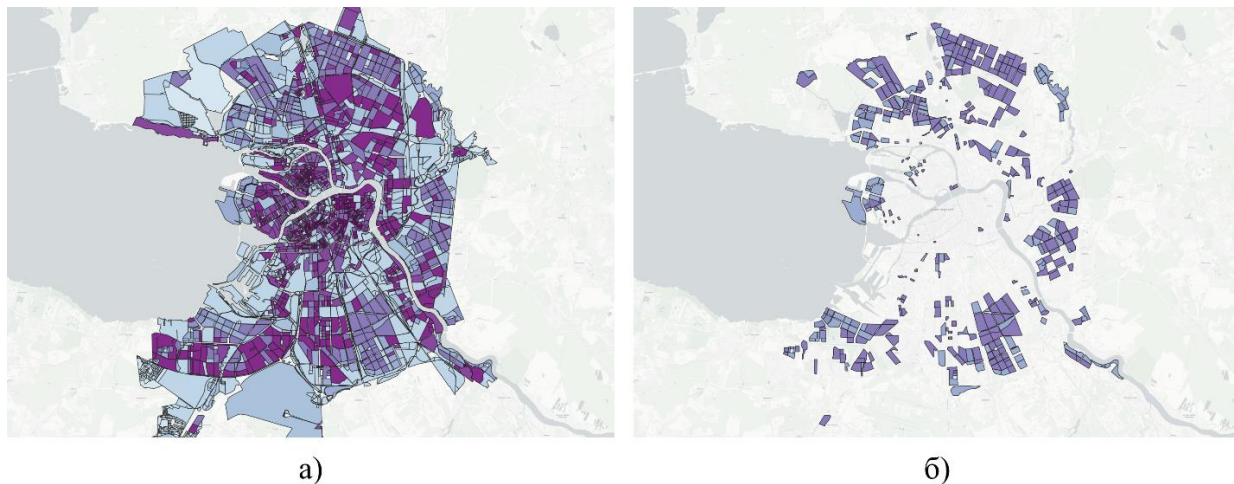


Рисунок 12 – а) Кварталы Санкт-Петербурга, посчитанные с помощью SpaceMatrix, б) выбранный морфотип

По завершении формирования тренировочных наборов данных была построена корреляционная иерархия генерируемых сервисов верхнего уровня (рисунок 13). После поэтапно реализованы части программного кода (Приложение 1 и 2).

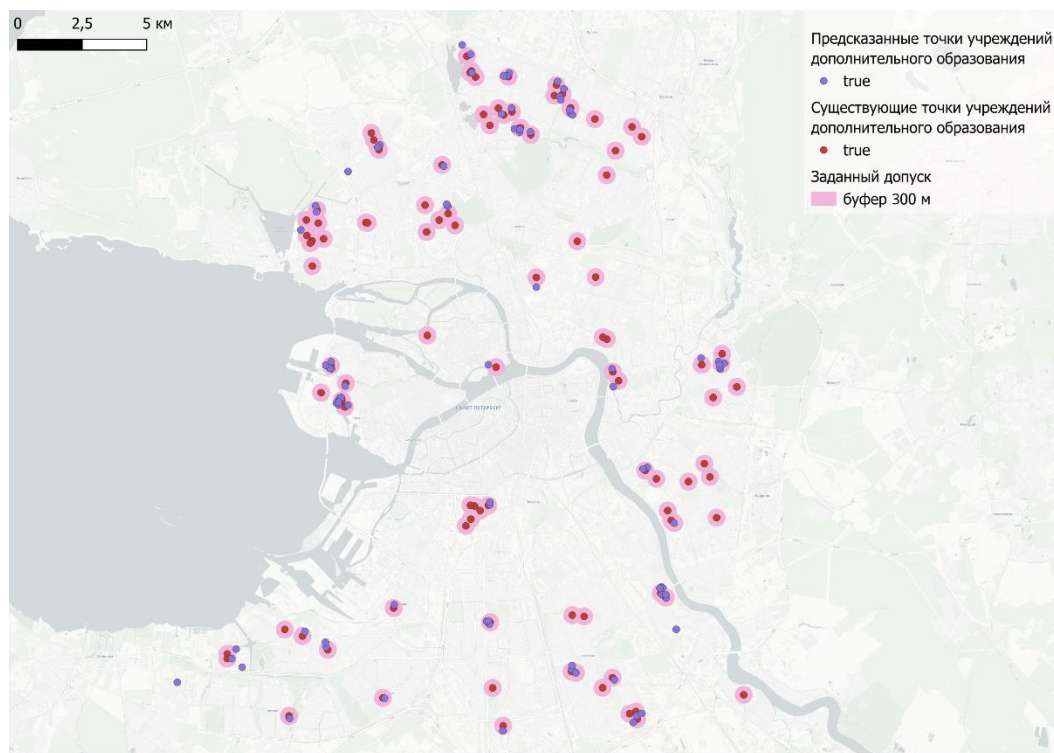


Рисунок 13 – Иерархия генерации сервисов верхнего уровня

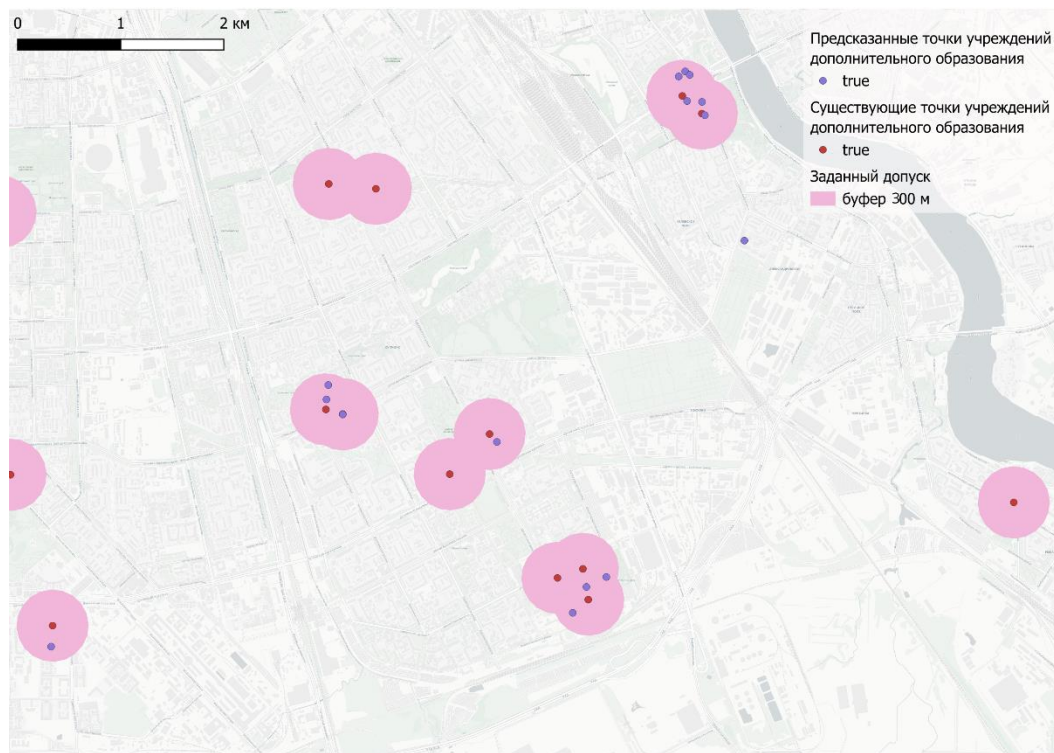
На Рисунке 14 представлены результаты первой итерации, демонстрирующей генерацию объектов дополнительного образования исключительно на основе первичных исходных данных.

Экспериментальная проверка выявила следующее: при наличии 30 реально существующих объектов дополнительного образования в тестовой выборке, модель точно идентифицировала местоположение 6 из них. Частичное соответствие с заданными

допусками продемонстрировано на Рисунке 14б. Стоит отметить, что доля случайных отклонений составила приблизительно 23%, что соответствует ожидаемым значениям для данного этапа разработки метода.



а)



б)

Рисунок 14 – Визуализация результатов первого этапа обучения модели: а) общая карта по городу, б) часть карты, где показаны заданные допуски и точки, попавшие в них

На Рисунке 15 представлена визуализация результатов второй итерации, в ходе которой осуществлялась генерация сервисов питания на основе первичных данных и ранее сгенерированных объектов дополнительного образования. Эмпирический анализ выявил пространственную зависимость между расположением объектов питания и такими элементами городской инфраструктуры, как жилые многоквартирные здания, объекты социального назначения и остановочные пункты транспортной сети [40].

В рамках данной итерации образовательные учреждения, выступавшие целевым параметром на предыдущем этапе, были интегрированы в модель в качестве одного из атрибутивных признаков. Посредством калибровки параметров алгоритма удалось достичь удовлетворительных показателей точности модели. Верификация на тестовой выборке, содержащей 198 реально существующих объектов питания, показала следующие результаты: модель корректно идентифицировала 75 объектов. Часть предложенных моделью точек соответствует установленным допускам, при этом уровень случайных отклонений не превышает 25%.

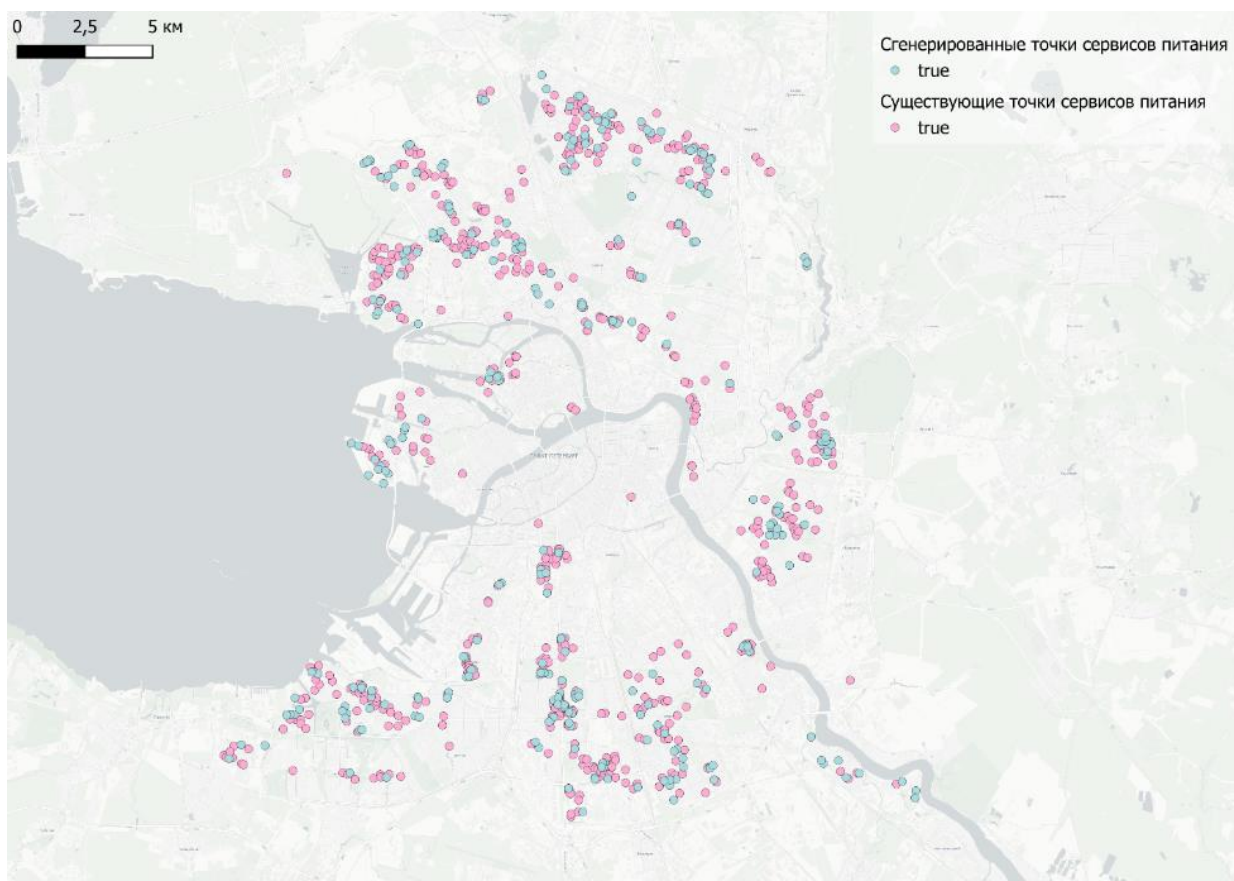


Рисунок 15 – Визуализация результатов второго этапа обучения модели

В результате итеративного процесса было обучено шесть классификационных моделей. Следующий шаг – применение обученных моделей к данным выбранной ранее территории ЖК «Солнечный город». Для этого разработан дополнительный скрипт, который последовательно применяет обученные модели к новым данным.



На Рисунке 16 показаны этапы генерации и распределения трех типов сервисов. В данных визуализациях здания отображены не полигональными объектами, а точечными центроидами (как при обучении):

- учреждения дополнительного образования (Рисунок 16б);
- сервисы питания (Рисунок 16в);
- торговые точки (Рисунок 16г).

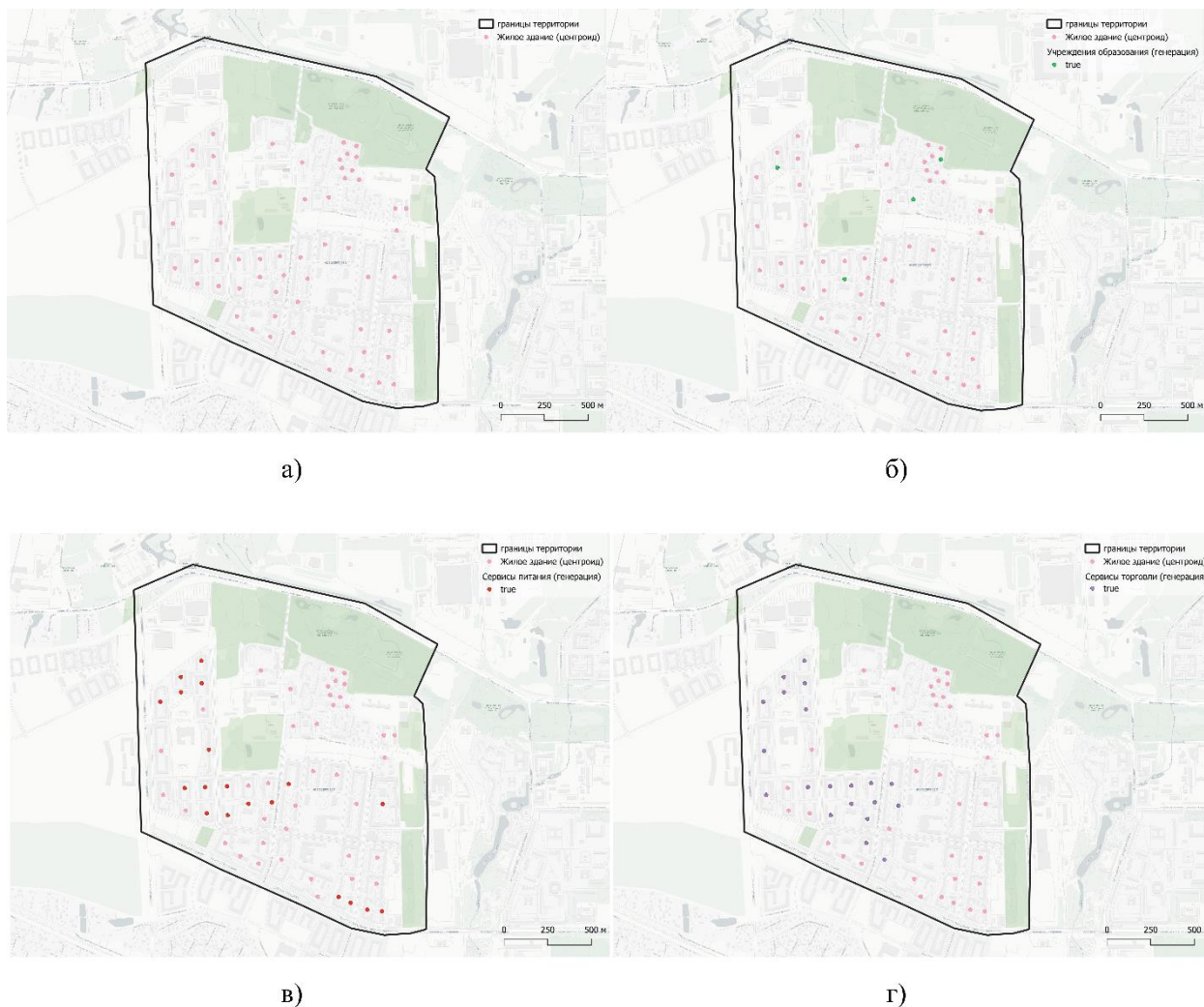


Рисунок 16 – Визуализация процесса генерации точек на трёх этапах верхнего уровня: а) учреждения дополнительного образования, б) сервисы питания, в) торговые точки

Для верификации эффективности данного этапа разработанного метода проведён сравнительный анализ между фактически существующей инфраструктурой на исследуемом участке и сгенерированными сервисными объектами, результаты которого визуализированы на Рисунке 17. Следует подчеркнуть, что исходные данные о реально существующих объектах получены посредством ручного картирования сервисов из системы 2GIS, поскольку открытые картографические данные OSM не содержат достаточной информации по рассматриваемой территории.



а)



б)



в)

Рисунок 17 – Сравнительный анализ существующей инфраструктуры и сгенерированных сервисов на исследуемом участке

Наблюдаемые несоответствия между смоделированными и фактическими объектами объясняются фрагментарностью данных OSM, что ограничивает возможности их использования в качестве референсного источника. В связи с этим оценке успешности экспериментального исследования следует основываться на расчётных метриках точности (Рисунок 18), обеспечивающих объективную количественную оценку соответствия между прогнозируемыми и реальными пространственными распределениями сервисов.

Рисунок 18 – Расчётные метрики точности: а) усреднённые метрики для 6 итераций обучения модели; б) усреднённые метрики для 6 итераций применения модели

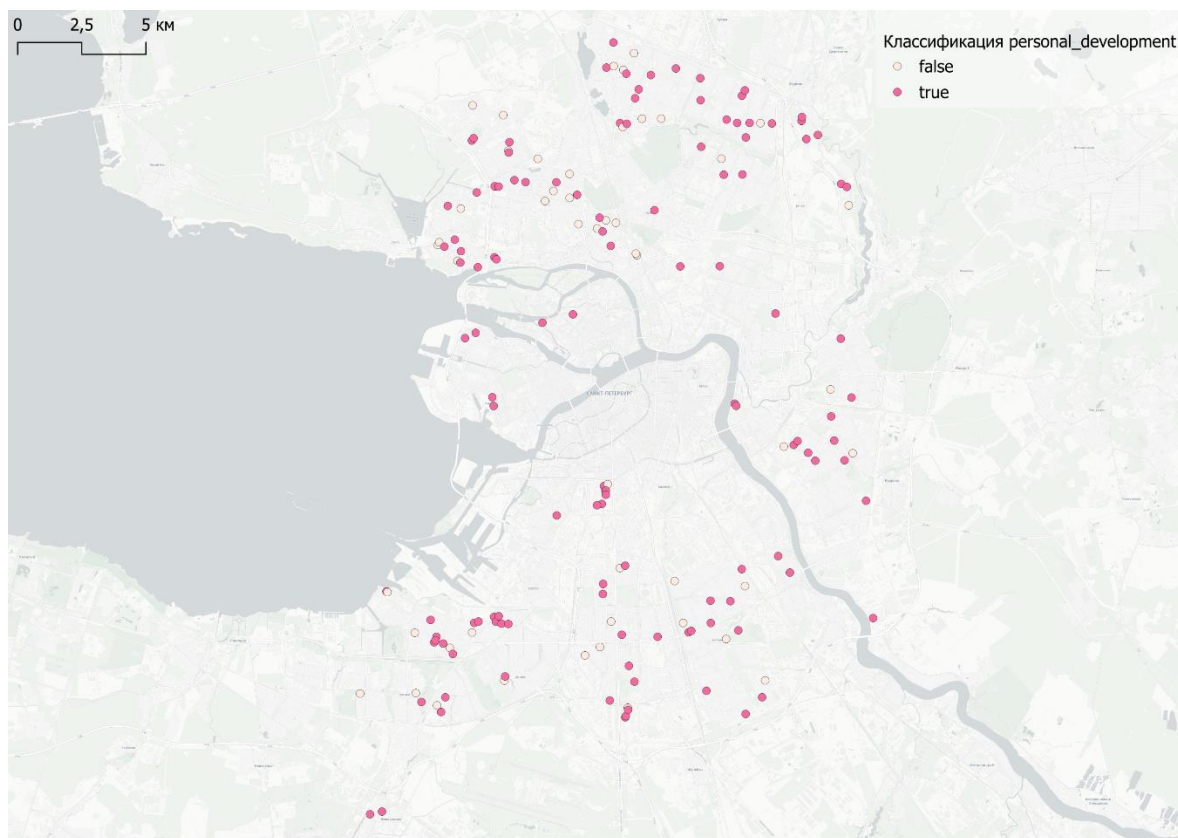
Для апробации второго уровня разработанного метода было принято решение выделить по две подгруппы в каждой укрупнённой группе. Тестирование проводилось на сложном с методологической точки зрения варианте – при наличии дисбаланса данных между подгруппами. Первая подгруппа характеризуется доминированием одного класса (мажорный класс), тогда как вторая объединяет несколько малых классов (минорный класс) в единую категорию.

"dancing\_school", "music\_school", "sport\_school", "language\_school", "library", "school", "kindergarten", "childcare". Выборка ограничивалась по признаку *education=1*.

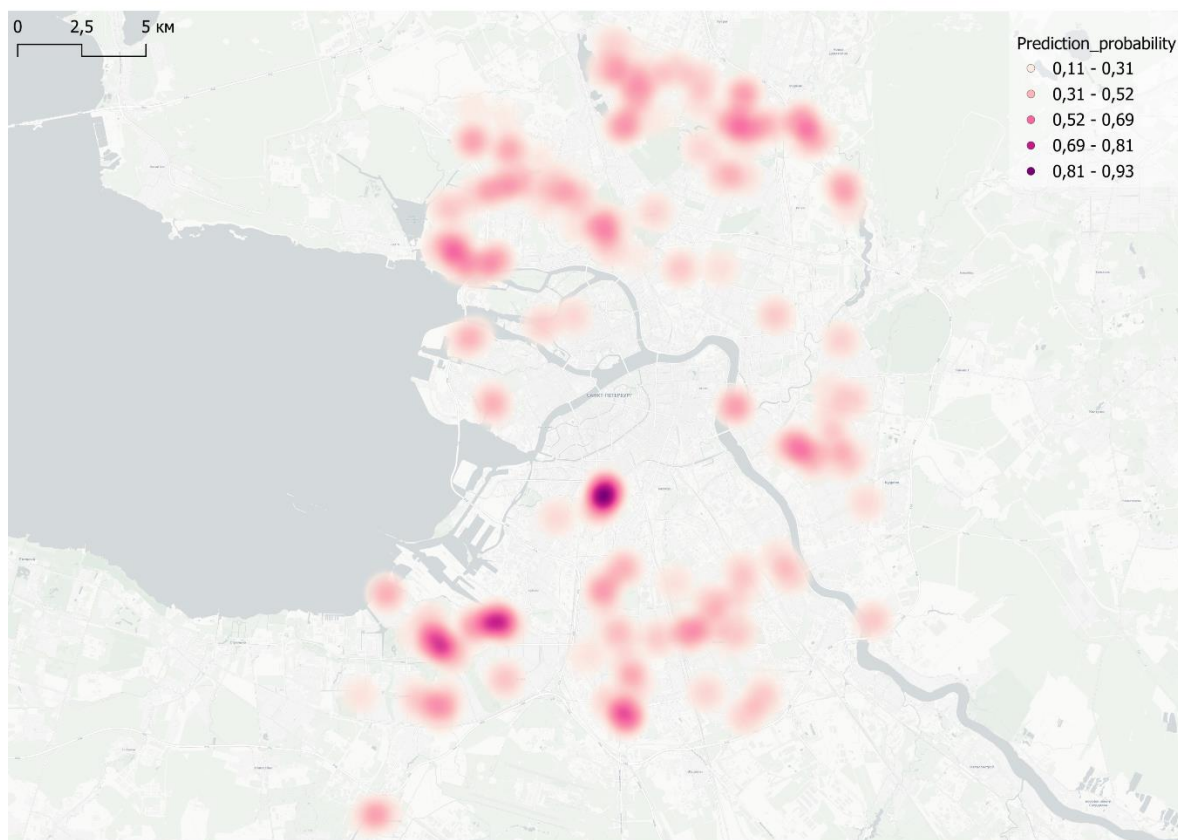
На Рисунке 19 представлены результаты обучения модели классификации для подгруппы *personal\_development* укрупнённой группы *education*. На Рисунке 19а отображены точки подгруппы *personal\_development*, классифицированные моделью, а на Рисунке 19б – тепловая карта, визуализирующая распределения вероятностей предсказания (*prediction\_probability*). Анализ точности модели показал достаточно высокие результаты, что свидетельствует о высокой обучаемости модели на задачах классификации и выявления устойчивых закономерностей в данных. Среднее значение метрики F1-score по 4 моделям составило 0,82. Для дальнейшей работы был выбран CatBoost.

На Рисунке 20 представлены результаты следующего итеративного этапа классификации минорной группы *else\_edu*. Исходный набор данных содержал 171 объект *education=1*, из которых 33 был отнесён к целевой подгруппе *else\_edu*. Несмотря на ограниченный объём выборки, модель продемонстрировала высокую эффективность классификации, что подтверждается значением метрики F1-score, составившим 0.84.



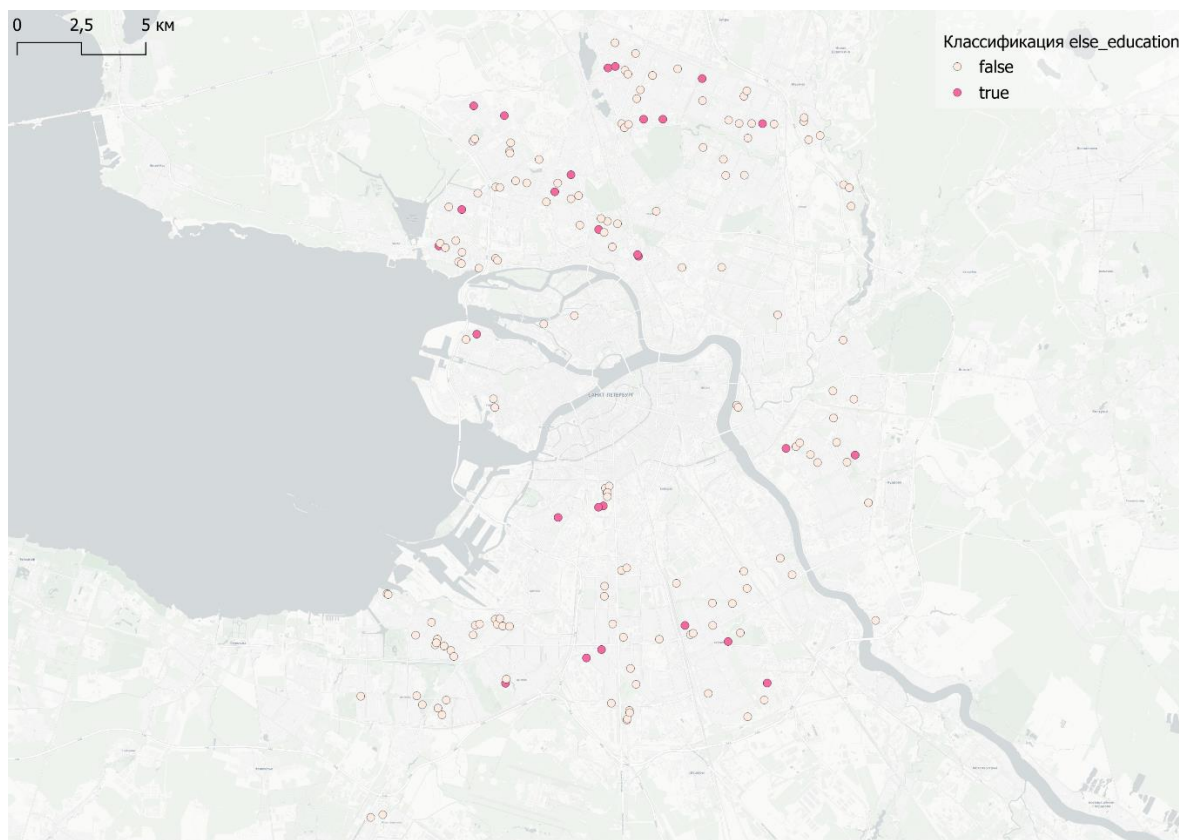


а)

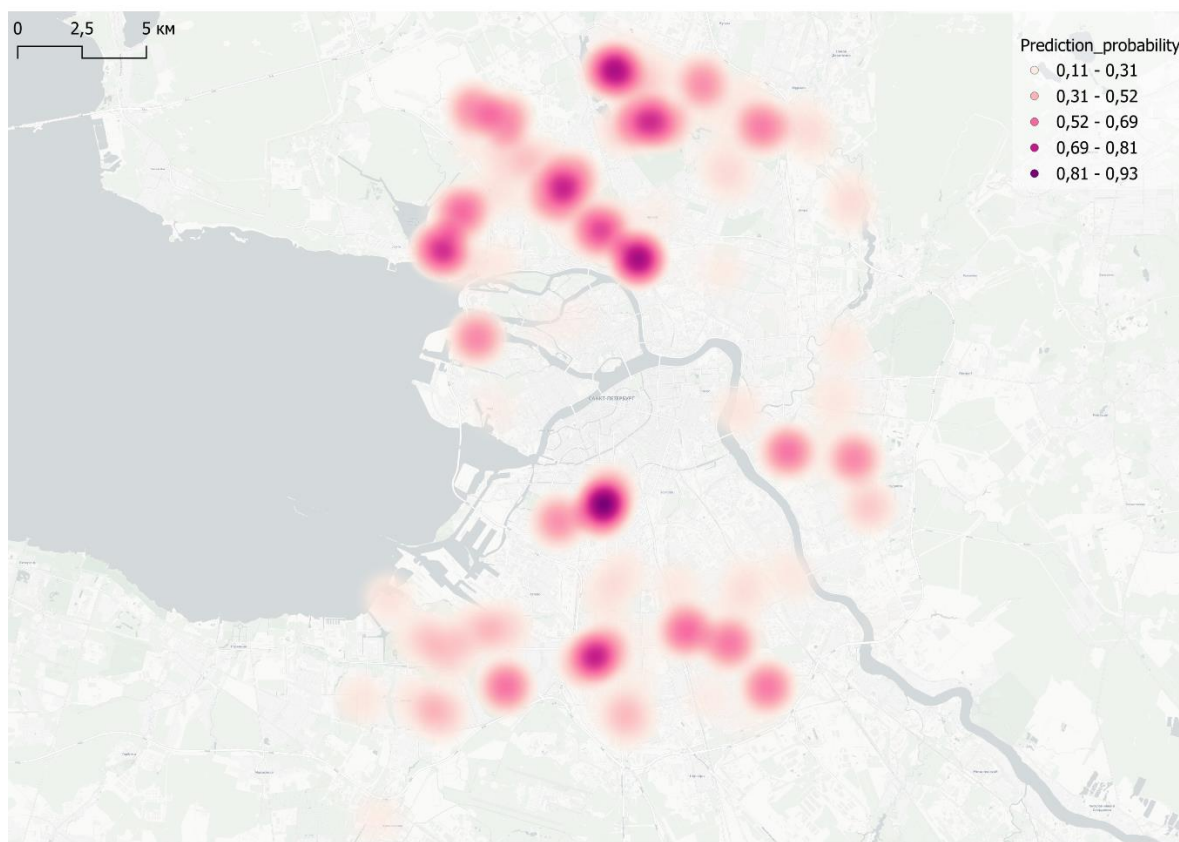


б)

Рисунок 19 – Результаты классификации подгруппы personal\_development укрупнённой группы education: а) точки подгруппы personal\_development, классифицированные моделью, б) тепловая карта распределения prediction\_probability



а)



б)

Рисунок 20 – Результаты обучения модели классификации для подгруппы else\_education укрупнённой группы education: а) точки подгруппы else\_education, классифицированные моделью, б) тепловая карта распределения prediction\_probability

Методология была последовательно применена ко всем описанным ранее укрупнённым группам. На Рисунке 21 представлена схема процесса классификации, отражающая иерархическое деление объектов исследования на подгруппы.

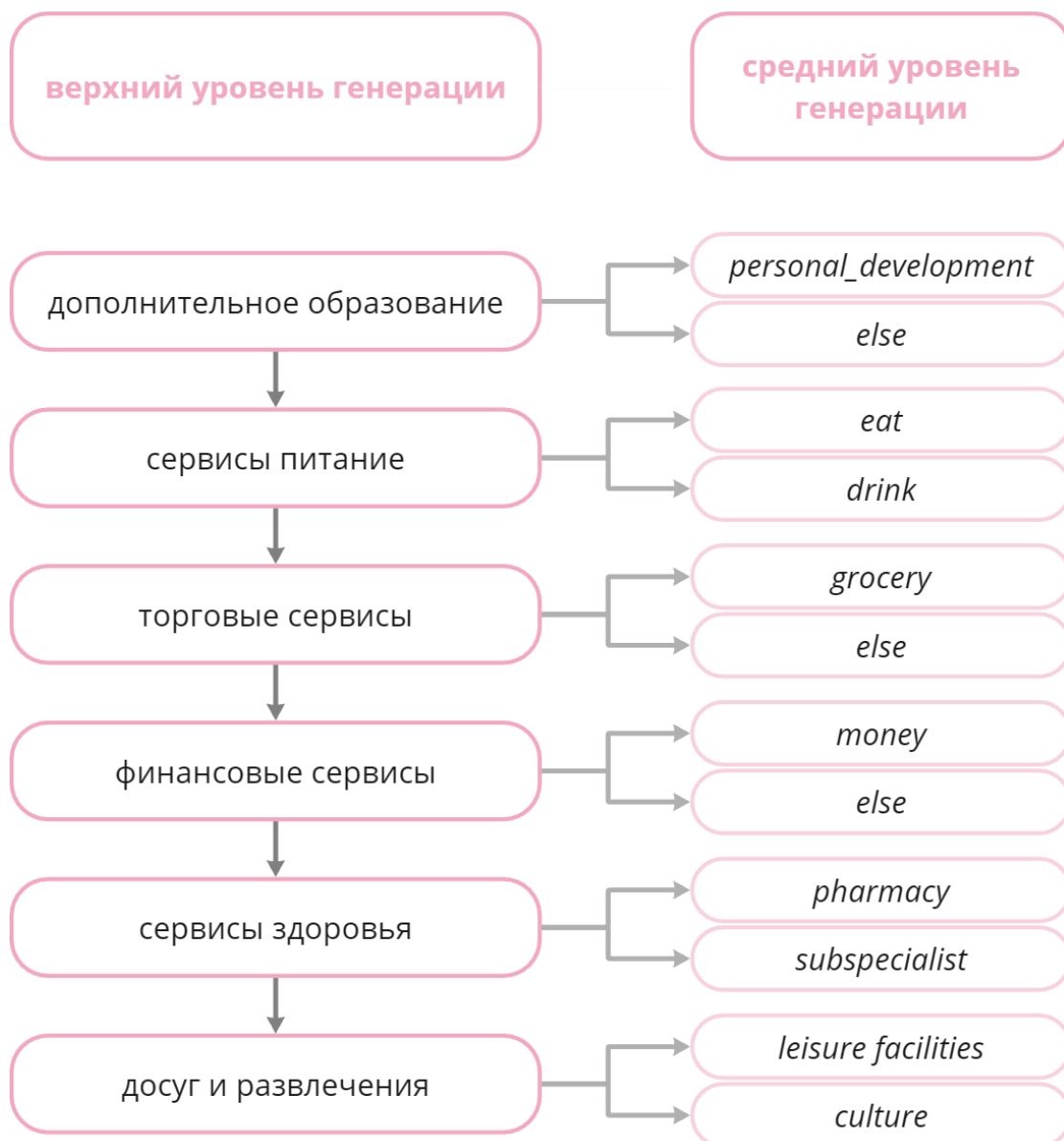
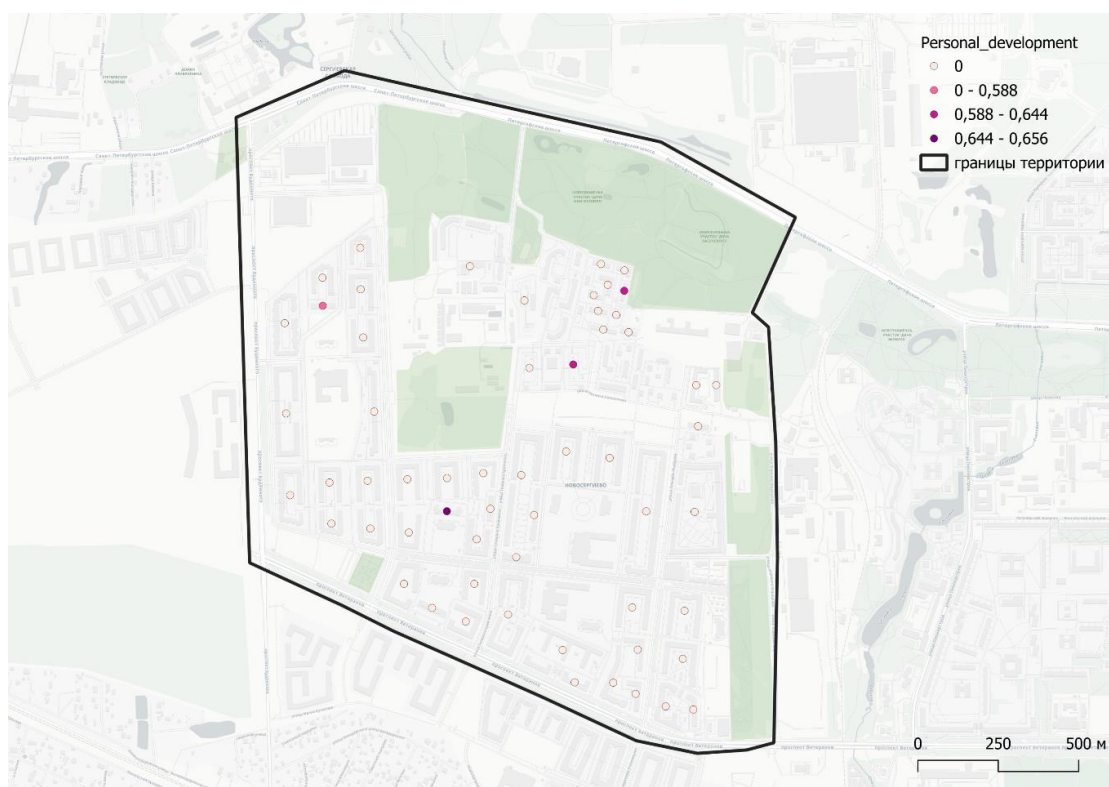


Рисунок 21 – Схема деления укрупнённых групп на подгруппы в контексте данного исследования

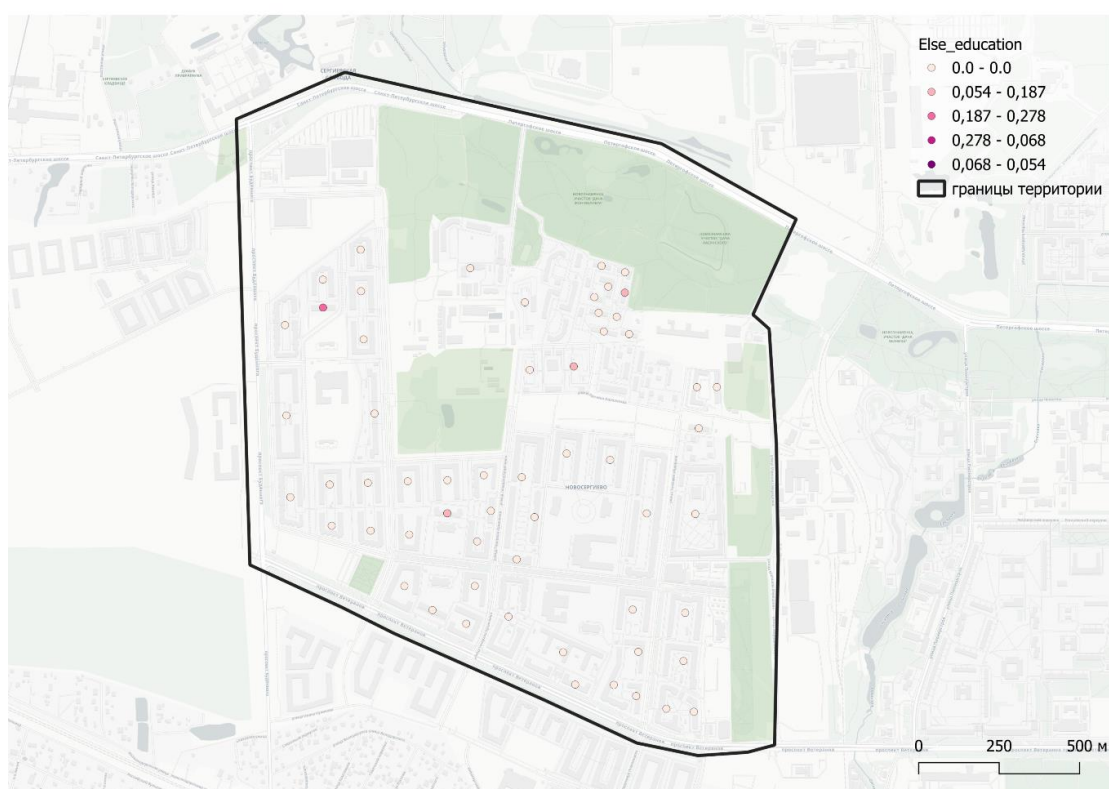
Следующий шаг – поэтапное применение обученных моделей к данным территории ЖК «Солнечный город». На рисунке 22 представлены результаты классификации сервисов дополнительного образования:

- итерация классификации подгруппы *personal\_development* (Рисунок 22а);
- итерация классификации подгруппы *else* (Рисунок 22б).





а)



б)

Рисунок 22 – Визуализация результатов применения модели классификации группы education: а) итерация классификации подгруппы *personal\_development*, б) итерация классификации подгруппы *else*

На рисунке 23 представлена визуализация результатов классификации сервисов питания:

- итерация классификации подгруппы *eat* (Рисунок 23а);
- итерация классификации подгруппы *drink* (Рисунок 23б).



а)



б)

Рисунок 23 – Визуализация результатов применения модели классификации группы *food*:  
а) итерация классификации подгруппы *eat*, б) итерация классификации подгруппы *drink*

### **3.1 Рекомендации к внедрению разработанного метода**

Предложенный метод может быть эффективно интегрирован в различные стадии городского планирования и развития территорий. Метод демонстрирует высокую степень адаптивности к изменяющимся условиям входных данных и может быть успешно применён для решения широкого круга исследовательских и прикладных задач девелоперами, градостроителями и участниками бизнеса.

Метод можно включить в процесс предпроектного анализа девелоперских проектов, реализуя цель оптимизации сервисного наполнения новых жилых районов. На этапе предпроектных исследований девелопер может использовать метод для определения оптимального набора коммерческих, социальных или иных типов сервисов (кафе, аптеки, фитнес-центры и др.), обоснования плотности и локализации точек притяжения на основе прогнозируемого спроса, а также минимизации рисков невостребованности коммерческих помещений (которые могут пустовать несколько лет после сдачи проекта).

Ещё одной сферой применения метода является возможность интегрировать его в проекты комплексного развития территорий, повышая эффективность их освоения за счёт обоснованного размещения сервисов. При разработке мастер-планов КРТ метод может помочь выявить «дефицитные» зоны, где отсутствуют ключевые сервисы, сформировать слоистую структуру сервисов (от макро- до микроуровня) и дополнительно обеспечить синергию между жилой застройкой, коммерцией и социальной инфраструктурой.

Метод также может быть использован для поддержки малого и среднего бизнеса при выборе локаций. Бизнес может использовать метод для выбора места открытия новой точки (магазин, кофейня, сервисный центр), оценки потенциала территории, а также анализа «точек роста» в районах с недостаточной обеспеченностью сервисами.

Для интеграции разработанного метода в градостроительную практику необходимо его согласование с действующей нормативной базой.

#### **Включение в предпроектный анализ девелоперских проектов.**

##### *Нормативная база:*

- Градостроительный кодекс РФ (ст. 48, 49) [16] – требует проведения предпроектных исследований при разработке документации по планировке территории (ППТ);
- СП 42.13330.2016 «Градостроительство. Планировка и застройка городских и сельских поселений» [17] – устанавливает требования к расчёту обеспеченности сервисами;
- ФЗ № 214 «Об участии в долевом строительстве» [48] – обязывает девелоперов обосновывать инфраструктурную насыщенность проектов.

##### *Механизмы внедрения:*

– Включение метода в раздел «Обоснование размещения объектов обслуживания» в составе ППТ, метод заменяет нормативные расчёты, основанные на условных допущениях, на подход, базирующийся на анализе данных (data-driven):

1. Сравнение с нормативами – модель проверяет, соответствует ли распределение сервисов СП 42.13330, и выявляет аномалии;

2. Согласование с Минстроем – результаты подаются как «экспериментальный расчёт на основе big data».

– Применение в качестве альтернативы укрупненным нормативам, которые часто устаревают и не учитывают локальные особенности, при согласовании с Минстроем:

1. Пилотные проекты – в отдельных районах согласовывается использование модели вместо СП 42.13330;

2. Сравнительный анализ – показывается, что data-driven подход снижает риски не востребоваемости объектов в силу обоснованного размещения в них сервисов;

3. Лоббирование изменений – результаты пилотов подаются в Минстрой для корректировки нормативов.

### **Интеграция в проекты комплексного развития территорий (КРТ).**

#### *Нормативная база:*

– ФЗ № 494 «О комплексном развитии территорий» [49] – регулирует вопросы КРТ;

– Методические рекомендации Минстроя по КРТ [50] – требуют анализа существующей и планируемой инфраструктуры;

– Стандарты ESG-трансформации городов (в т.ч. ГОСТ Р 58841-2020) [51] – рекомендуют data-driven подходы к планированию.

#### *Механизмы внедрения:*

– Включение метода в «Схему функционального зонирования» градостроительной документации (ППТ, КРТ, Генплан) в качестве инструмента обоснованного распределения сервисов для учитывания реальных паттернов спроса и взаимодействия сервисов:

1. Анализ существующей застройки и спроса:

а) На этапе подготовки схемы метод применяется для анализа сложившейся структуры сервисов и выявления зон дефицита или избытка;

б) Используется макромасштабная генерация с учётом данных о населении, транспортной доступности и существующей инфраструктуре.

2. Корректировка зонирования

а) Результаты модели – распределение точек притяжения – накладываются на проектную схему;

б) Зоны с высокой прогнозируемой активностью коммерческих сервисов выделяются как «смешанные» или «общественно-деловые»;

в) Уточняются границы жилых зон с учетом пешеходной доступности сервисов.

### 3. Нормативное закрепление

а) Предложения по зонированию обосновываются расчётами модели, что снижает субъективность решений;

б) В пояснительную записку включается раздел «Анализ сервисного наполнения на основе данных и ML».

– Использование для обоснования «точек роста» в рамках публичных слушаний путём предоставления доказательной базы для обоснования решений о развитии территорий, что потенциально может повысить доверие участников слушаний (жителей, инвесторов, регуляторов):

1. Выявление потенциальных локаций – на основе данных о УДС, плотности населения и существующих сервисах модель выделяет зоны, с высокой вероятностью подходящие под необходимый тип бизнеса;

2. Визуализация и обоснование;

3. Публичная защита – данные модели используются как аргумент при обсуждении.

### **Поддержка малого и среднего бизнеса при выборе локаций.**

*Нормативная база:*

– ФЗ № 209 «О развитии малого и среднего предпринимательства» – предусматривает меры поддержки бизнеса [52];

*Механизмы внедрения:*

– Разработка муниципальных «карт бизнес-возможностей» (инструмент для предпринимателей, показывающий, какие сервисы востребованы в конкретных районах) на основе метода (как часть программ поддержки МСП):

1. Формирование рекомендаций – на основе мезо- и микромасштабной генерации определяются типы сервисов с высокой prediction\_posibilities (например, аптеки в районе с пожилым населением), оптимальные;

2. Публикация в открытом доступе

а) Данные интегрируются в интерактивную карту;

б) Для каждого участка указываются рекомендуемые виды деятельности, конкуренция (количество существующих аналогичных сервисов).

3. Связь с мерами поддержки

а) Льготная аренда муниципальных помещений в «приоритетных» зонах;

б) Упрощенное согласование для бизнеса, соответствующего рекомендациям.



– Включение в «Инвестиционные паспорта» территорий, предоставляемые бизнесу:

1. Анализ территории;
2. Формирование раздела «Рекомендации по размещению» – в паспорт включаются карты с зонированием по типам сервисов;
3. Продвижение среди инвесторов.

## ЗАКЛЮЧЕНИЕ

В ходе проведённого исследования была разработана метод пространственного размещения точек притяжения на слабоурбанизированных территориях, основанный на комбинации графовых нейронных сетей и supervised machine learning. Метод реализуется итеративно – через трёхуровневую систему генерации.

Актуальность работы обусловлена недостаточной эффективностью традиционных нормативных подходов при планировании новых районов и территорий комплексного развития, где отсутствует достаточный объём данных, но требуется пространственное распределение сервисов и объектов инфраструктуры.

Основные результаты исследования заключаются в следующем:

- Проведён комплексный анализ существующих методов градостроительного планирования, выявлены их ограничения применительно к территориям с недостаточным наличием данных;
- Разработан трёхуровневый метод генерации точек притяжения (макро-, мезо-, микромасштаб) с использованием алгоритмов машинного обучения;
- Обоснована эффективность графовых нейронных сетей и Supervised machine learning для выявления пространственных взаимосвязей, не учитываемых в нормативных подходах, что повышает точность прогнозирования оптимального расположения объектов;
- Методология апробирована на тестовой территории, что подтвердило её способность снижать дисбалансы в распределении сервисов и повышать качество городской среды и её разнообразие;
- Сформулированы практические рекомендации по внедрению метода в градостроительные практики, включая адаптацию под различные типы территорий и интеграцию с существующими системами планирования.

Проведённое исследование подтвердило гипотезу о том, что применение машинного обучения, в частности графовых нейронных сетей и Supervised machine learning, позволяет минимизировать неопределённость при проектировании новых районов, способствуя созданию более устойчивой и комфортной городской среды.

Таким образом разработанный метод является достаточно эффективным инструментом пространственного распределения точек притяжения, обладает хорошим потенциалом для оптимизации и расширения возможностей генерации. Перспективы дальнейших исследований связаны с расширением набора учитываемых параметров (например, мобильность населения, экологические факторы), а также с разработкой автоматизированных инструментов для поддержки принятия решений в urban planning.

Внедрение разработанного метода в практику градостроительства может стать важным шагом в развитии умного и устойчивого планирования городских территорий.

Выполненное исследование и полученные результаты соответствуют направлению подготовки 07.04.04 Градостроительство.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Резолюция, принятая Генеральной Ассамблеей 25 сентября 2015 года. Преобразование нашего мира: Повестка дня в области устойчивого развития на период до 2030 года. [Электронный ресурс]. URL: <https://documents-dds-ny.un.org/doc/UNDOC/GEN/N15/291/92/PDF/N1529192.pdf?OpenElement> (Дата обращения: 01.11.2023)
2. ЦУР 11. Устойчивые города и населённые пункты. [Электронный ресурс]. URL: <https://sdg.openshkola.org/goal11#targets> (Дата обращения: 01.11.2023)
3. Синицына И. А. Адаптация инструментов пространственного синтаксиса для анализа освоения территории // *Architecture and Modern Information Technologies*. – 2022. – №2 (59). – С. 129–141.
4. Шалина Д. С., Тихонов В. А. Создание точек притяжения горожан для повышения привлекательности района Пионерский в Екатеринбурге // *Академический Вестник УРАЛНИИПРОЕКТ РААСН*. – 2022. – №4. – С. 34–39.
5. Суслов С. И., Гаврикова К. В. Места притяжения граждан в городском пространстве на примере Красногвардейского района Санкт-Петербурга // *Общество: политика, экономика, право*. – 2017. – № 5. – С. 17–19.
6. Громенко И. В., Гарнага А. Ф., Охотникова Ю. В. и др. Реновация площади Блюхера в городе Хабаровске как средство создания комфортной городской среды // *Инновации и инвестиции*. – 2020. – № 12. – С. 177–181.
7. Тяглов С. Г., Родионова Н. Д., Федорова Я. В., Сергиенко В. Ю. Алгоритм развития общественных пространств крупных городов в условиях их урбанизации // *Регионология*. – 2020. – № 4 (113). – С. 778–800.
8. Амосова Е. В. Творческие кластеры как способ сохранения и редевелопмента объектов индустриального наследия // *Общество. Среда. Развитие. (Terra Humana)*. – 2019. – Том XXVI. – №1 – С. 224–256.
9. Лавруевич П.Е. Скалабан И.А. Ценность места: как воспроизводится символическая собственность в стихийной рекреации (кейс Караканского бора) // *Журнал социологии и социальной антропологии*. – 2023. – № 3. – С. 98–102.
10. Cai, L., Jiang, F., Zhou, W., Li, K. Design and Application of an Attractiveness Index for Urban Hotspots Based on GPS Trajectory Data // *IEEE Access*. – 2018. – №6. – P. 55976–55985.
11. Alhazzani, M., Alhasoun, F., Alawwad, Z., Gonza'lez, M.C. Urban attractors: Discovering patterns in regions of attraction in cities // *PLoS ONE*. – 2021. – №16 (4).

12. Санкт-Петербург для жизни и для развлечений. [Электронный ресурс]. URL: <https://yandex.ru/company/researches/2017/spb/districts> (Дата обращения: 30.11.2023)
13. Баталина Т.С. Анализ особенностей формирования общественного пространства // Бизнес и дизайн ревю. – 2017. – Т. 1. – № 1 (5). – С. 11.
14. Стандарт комплексного развития территорий. [Электронный ресурс]. URL: <https://xn--d1aqf.xn--p1ai/urban/standards/printsipy-kompleksnogo-razvitiya-territoriy/> (Дата обращения: 25.11.2023)
15. Lee C. The Challenges of Service Planning in New Urban Areas // Journal of Urban Development. – 2023. – Vol. 12. – P. 45–60.
16. «Градостроительный кодекс Российской Федерации» от 29.12.2004 N 190-ФЗ (ред. от 13.06.2023). [Электронный ресурс]. URL: [https://www.consultant.ru/document/cons\\_doc\\_LAW\\_51040/](https://www.consultant.ru/document/cons_doc_LAW_51040/) (Дата обращения: 01.11.2023)
17. СП 42.13330.2016 «Градостроительство. Планировка и застройка городских и сельских поселений». [Электронный ресурс]. URL: <https://docs.cntd.ru/document/456054209> (Дата обращения: 01.11.2023.)
18. Яргина З.Н., Хачатрянц К.К. Социальные основы архитектурного проектирования: учеб. для архитектур. Вузов. – М.: Стройиздат, 1990. – С. 343.
19. Орлова А. А., Задвернюк Л. В. Современные методы и средства анализа городской среды // Новые идеи нового века: материалы международной научной конференции ФАД ТОГУ. – Федеральное государственное бюджетное образовательное учреждение высшего образования Тихоокеанский государственный университет, 2018. – Т. 1. – С. 338-342.
20. Первое знакомство с полносвязными нейросетями. [Электронный ресурс]. URL: <https://education.yandex.ru/handbook/ml/article/pervoe-znakomstvo-s-polnosvyaznymi-nejrosetyami> (Дата обращения: 05.12.2024)
21. Jin G., Sha H., Feng Y., Cheng Q., Huang J. GSEN: An Ensemble Deep Learning Benchmark Model for Urban Hotspots Spatiotemporal Prediction // Neurocomputing. – 2021. – Vol. 455 – P. 353-367.
22. Гринева Н. В., Иванов И. Д., Аблязина Н. Х. Интерпретируемая модель машинного обучения для задачи геоаналитики: моделирования размещения торговых точек розничной сети // Инновации и инвестиции. – 2023. – №6.
23. Delhoum, Y.; Belaroussi, R. Modeling Visit Potential to Predict Hotspots of a Future District. // Infrastructures. – 2023. – №8 (10) – 145.
24. Lynch, K. Good City Form; MIT Press: Cambridge, MA, USA, 1984.

25. Herthogs, P.; Tunçer, B.; Schläpfer, M.; He, P. A Weighted Graph Model to Estimate People's Presence in Public Space-The Visit Potential Model. // In Proceedings of the eCAADe 2018—36th Annual Conference, Lodz, Poland. – 2018.
26. Качинкина Е. А., Устинова Ю. С. Основные методы сбора данных // Теория и практика современной науки. – 2020. – №. 12 (66). – С. 117-126.
27. Мазур, Е. В., Бабешко В. Н. Парсинг как источник получения данных для анализа // Перспективное развитие науки, техники и технологий : Сборник научных статей 9-ой Международной научно-практической конференции. – 2019. – С. 161-164.
28. Самохвалов Ю. Я., Науменко Е. М. Экспертное оценивание // Методический аспект / ЮЯ Самохвалов, ЕМ Науменко–К.: ДУКИТ. – 2007.
29. Дугушкина, Н. В. Обзор популярных методов кластеризации в машинном обучении // Наукосфера. – 2020. – № 7. – С. 112-118.
30. Шумилов А. В. Оценивание гравитационных моделей международной торговли: обзор основных подходов // Экономический журнал ВШЭ. 2017. – №2. – С. 224-248.
31. Кольцов С. Н. Регрессионный анализ. [Электронный ресурс]. URL: <https://www.hse.ru/data/2014/08/29/1313619461/%D0%BB%D0%B5%D0%BA%D1> (Дата обращения: 12.05.2024)
32. Петухов К. В., Стригунов Ю.В. Нейросетевые модели и анализ эффективности их применения для оценки и прогнозирования показателей // Современные проблемы и пути их решения в науке, производстве и образовании. – 2017. – № 1. – С. 73-77.
33. Белоглазов Д. А. Особенности нейросетевых решений, достоинства и недостатки, перспективы применения // Известия ЮФУ. Технические науки. – 2008. – №7. – С. 105-110
34. Черкасов Д. Ю., Иванов В. В. Машинное обучение // Наука, техника и образование. – 2018. – №5 (46).
35. Введение в машинное обучение и анализ данных. [Электронный ресурс]. URL: [https://courses.openedu.ru/courses/course-v1:ITMOUniversity+INTROMLAAISB+spring\\_2024\\_ITMO\\_mag/pdfbook/0/](https://courses.openedu.ru/courses/course-v1:ITMOUniversity+INTROMLAAISB+spring_2024_ITMO_mag/pdfbook/0/) (Дата обращения: 12.05.2024)
36. Механизмы разработки и внедрения мастер-плана. [Электронный ресурс]. URL: [https://itpgrad.ru/upload/iblock/3f5/xewoql3qo18vzox8y6fn1ug99s19oi63/Natsionalnyy-standart-master\\_planov\\_Kniga3.pdf](https://itpgrad.ru/upload/iblock/3f5/xewoql3qo18vzox8y6fn1ug99s19oi63/Natsionalnyy-standart-master_planov_Kniga3.pdf) (Дата обращения: 13.12.2024)
37. Градостроительная деятельность и архитектура. [Электронный ресурс]. URL: <https://minstroyrf.gov.ru/trades/gradostroitelnaya-deyatelnost-i-arhitektura/> (Дата обращения: 13.12.2024)

38. Библиотека с открытым кодом BlocksNet. [Электронный ресурс]. URL:<https://aimclub.github.io/blocksnet/> (Дата обращения: 10.12.2024)
39. Системы менеджмента качества. Основные положения и словарь. [Электронный ресурс]. URL: <https://rustestm.ru/wp-content/uploads/2021/10/gost-r-iso-9000-2015-sistemy-menedzhmenta-kachestva-osnovnye-polozeniya-i-slovar.pdf> Дата обращения: 16.12.2024)
40. Chen Y. et al. Understanding the spatial organization of urban functions based on co-location patterns mining: A comparative analysis for 25 Chinese cities // *Cities*. – 2020. – Т. 97. – С. 102563.
41. Tuning the hyper-parameters of an estimator. [Электронный ресурс]. URL:<https://aimclub.github.io/blocksnet/> (Дата обращения: 16.12.2024)
42. Jacobs J. *The Death and Life of Great American Cities*. — New York: Random House, 1961.
43. Gehl J. *Cities for People*. — Washington: Island Press, 2010.
44. Nadkarni S., Prügl R. Digital transformation: a review, synthesis and opportunities for future research // *Small Business Economics*. 2021. – Vol. 57. – P. 203-223.
45. Хиллиер Б. Пространство – это машина: Конфигурационная теория архитектуры / пер. с англ. А. Соколова. – М.: Strelka Press, 2015. — 464 с.
46. Kipf T. N., Welling M. Semi-Supervised Classification with Graph Convolutional Networks // *Proceedings of the 5th International Conference on Learning Representations (ICLR)*. — 2017.
47. Yuan Y., Chen X., Li W. Spatial optimization of retail locations based on pedestrian accessibility metrics using machine learning // *Computers, Environment and Urban Systems*. 2022. – Vol. 92. – P. 101749.
48. Федеральный закон от 30.12.2004 № 214-ФЗ. [Электронный ресурс]. URL: [http://www.consultant.ru/document/cons\\_doc\\_LAW\\_51099/](http://www.consultant.ru/document/cons_doc_LAW_51099/) (Дата обращения: 23.03.2025).
49. Федеральный закон от 31.07.2020 № 494-ФЗ. [Электронный ресурс]. URL: <http://publication.pravo.gov.ru/Document/View/0001202007310036> (Дата обращения: 23.03.2025).
50. Методические рекомендации по комплексному развитию территорий. [Электронный ресурс]. URL: <https://minstroyrf.gov.ru/docs/> (дата обращения: 23.03.2025).
51. ГОСТ Р 58841-2020. [Электронный ресурс]. URL: <https://www.gost.ru/> (Дата обращения: 23.03.2025).
52. Федеральный закон от 24.07.2007 № 209-ФЗ. [Электронный ресурс]. URL: <https://base.garant.ru/12154854/> (Дата обращения: 23.03.2025).

53. Градостроительный кодекс Российской Федерации от 29.12.2004 № 190-ФЗ (ред. от 24.03.2021). [Электронный ресурс] URL: [http://www.consultant.ru/document/cons\\_doc\\_LAW\\_51040/](http://www.consultant.ru/document/cons_doc_LAW_51040/) (дата обращения: 20.02.2024).

54. Smith A., Brown B. Urban Planning in Developing Areas. 2nd Edition. Routledge, 2020. 345 p.



## ПРИЛОЖЕНИЕ 1

### Первая часть кода для работы в графовой нейронной сети (построение графа)

```
import json
import pickle
from pathlib import Path

import networkx as nx
import numpy as np
import geopandas as gpd
import pandas as pd
from shapely.geometry import Point
import matplotlib.pyplot as plt
from geopy.distance import distance
import plotly.graph_objects as go

buffer_size = 300
n_closest = 3

path_root_city = Path('data/input/spb')

filename = 'data_high.csv'

filepath = path_root_city / filename

filepath_subway = path_root_city / 'метро.geojson'
filepath_bus = path_root_city / 'остановки автобусов.geojson'
filepath_tram = path_root_city / 'остановки трамвай.geojson'

df_squares = gpd.read_file(path_root_city / 'squares.csv')
df_squares['fid'] = df_squares['fid'].astype(int)

def read_geojson(path: Path):
    with open(path, 'r', encoding='utf-8') as f:
        data = json.load(f)

    return data

def extract_stops(path: Path) -> list[tuple[int, float, float]]:
    data = read_geojson(path)

    coords = list(map(lambda v: (int(v['properties']['osm_id']),
                                  v['geometry']['coordinates'][0],
                                  v['geometry']['coordinates'][1]),
                       data['features']))
    return coords
```

```

def draw_plotly(g_):
    import plotly.express as px

    pos_ = nx.get_node_attributes(g_, 'coordinate')

    nodes = list(g_.nodes())

    edge_x = []
    edge_y = []
    edges_text = []
    for edge in g_.edges(data=True):
        x0, y0 = pos_[edge[0]]
        x1, y1 = pos_[edge[1]]

        edge_x.extend([x0, x1, None])
        edge_y.extend([y0, y1, None])
        edges_text.extend([f'Meters: {edge[2]['meters']}', f'Meters:
{edge[2]['meters']}', None])

    nodes_data = g_.nodes(data=True)
    node_x = []
    node_y = []
    node_text = []
    node_colors = []
    for node in nodes:
        x, y = pos_[node]
        node_x.append(x)
        node_y.append(y)
        attrs = nodes_data[node]
        if attrs['label'] in {'subway', 'bus', 'tram'}:
            node_colors.append('black')
        else:
            node_colors.append('green' if attrs['is_flower'] > 0 else 'red')

        attrs = dict(filter(lambda kv: kv[0] not in {'coordinate', 'label'},
attrs.items()))
        node_text.append('<br>'.join(list(map(lambda kv: f'{kv[0]}: {kv[1]}',
attrs.items()))))

    fig = go.Figure()
))

df_nodes =
pd.DataFrame.from_records(list(dict(nodes_data).values())).drop(columns='coor
dinate')

df_nodes['s_cnt'] = df_nodes[['bar', 'pab', 'cafe', 'fastfood', 'resto',
'bakery', 'beaty', 'hair', 'shoes', 'hotel']].sum(axis=1)

df_nodes['color'] = 'gray'
df_nodes.loc[df_nodes['s_cnt'] > 0, 'color'] = 'blue'
df_nodes.loc[df_nodes['is_flower'] == 1, 'color'] = 'green'

```

```

df_nodes.loc[df_nodes['label'].isin({'tram', 'subway', 'bus'}), 'color']
= 'black'

fig = px.scatter_map(df_nodes, lat="lon", lon="lat",
hover_name="is_flower", hover_data=["label", "square"],
                    color=df_nodes['color'],
                    zoom=8,
                    height=1000)

fig.update_layout(map_style="open-street-map")
fig.update_layout(margin={"r": 0, "t": 0, "l": 0, "b": 0})

fig.update_layout(
    title='Interactive Graph with NetworkX and Plotly',
    hovermode='closest',

fig.show()

def calculate_distances(geometry: gpd.GeoSeries, point: tuple[float, float])
-> pd.Series:
    dst = gpd.GeoSeries([Point(point), ], crs='epsg:4326').to_crs(epsg=3857)
    return geometry.to_crs(epsg=3857).distance(dst[0])

def add_stops(
    g_: nx.Graph,
    positions: dict[int, tuple[float, float]],
    stops: list[tuple[int, float, float]],
    label: str
):
    geometry = [Point(xy) for xy in list(positions.values())]
    gdf_pos = gpd.GeoDataFrame(list(zip(positions.keys(),
positions.values()))), columns=['id', 'geometry'], crs='epsg:4326',
geometry=geometry)

    for stop_id, lat, lon in stops:
        distances = calculate_distances(gdf_pos.geometry, (lat, lon))

        distances_less = distances[distances < buffer_size]
        if distances_less.shape[0] > 0:
            if g_.has_node(stop_id):
                raise ValueError('The node already exists')

            g_.add_node(stop_id, coordinate=(lat, lon),
                        label=label, bar=0, pab=0, cafe=0, fastfood=0,
resto=0,
                        bakery=0, beaty=0, cloth=0, hair=0, shoes=0,
is_flower=0,
                        lat=lat, lon=lon)

        for building_id in gdf_pos[distances < buffer_size]['id']:
            g_.add_edge(building_id, stop_id, meters=0)

```

```

def build_data():
    cache_file_path = Path('data') / filename.replace('.csv', '.pkl')
    if cache_file_path.exists():
        with open(cache_file_path, 'rb') as f:
            return graph

    df = pd.read_csv(filepath)
    data_json = {'features': df.to_dict(orient='records')}

    stops_bus = extract_stops(filepath_bus)
    stops_tram = extract_stops(filepath_tram)
    stops_sub = extract_stops(filepath_subway)

    graph = nx.Graph()

    squares_map = df_squares.set_index('fid')['area'].to_dict()

    n_count = len(data_json['features'])
    for i, building in enumerate(data_json['features']):
        if i % 100 == 0:
            print(f'Calc {i} / {n_count}')

        props = building['properties'] if 'properties' in building else
building
        building_id = props['building_id']

        square = float(squares_map[building_id])
        is_flower = int(props['цветочные'] > 0)
        if square < 200:
            print(f'Skip building "{building_id}" with square {square} and
is_flower={is_flower}')
            continue

        x_source, y_source = building['x'], building['y']

        graph.add_node(props['building_id'], coordinate=(x_source, y_source),
                        square=square,
                        label='Y' if props['цветочные'] else 'N',
                        bar=props['бары'],
                        pab=props['пабы'],
                        cafe=props['кафе'],
                        fastfood=props['фастфуд'],
                        resto=props['рестораны'],
                        bakery=props['выпечка'],
                        beauty=props['салоны'],
                        cloth=props['одежда'],
                        hair=props['парикмахерские'],
                        shoes=props['обувь'],
                        hotel=props['отели'],
                        is_flower=is_flower,
                        lat=x_source,

```

```

        lon=y_source
    )

    print('The nodes was build')

    pos_: dict[int, tuple[float, float]] = nx.get_node_attributes(graph,
'coordinate')
    geometry = [Point(xy) for xy in list(pos_.values())]
    gdf_pos = gpd.GeoDataFrame(list(zip(pos_.keys(), pos_.values()))),
columns=['id', 'geometry'], crs='epsg:4326', geometry=geometry)

    n_count = len(pos_)
    for i, (building_id, coord) in enumerate(pos_.items()):
        if i % 100 == 0:
            print(f'Calc {i} / {n_count}')

        gdf_pos['distances'] = calculate_distances(gdf_pos.geometry, coord)

        indices = (gdf_pos['distances'] > 0) & (gdf_pos['distances'] <
buffer_size)
        if indices.any():
            gdf_close = gdf_pos[gdf_pos['distances'] < buffer_size]
        else:
            gdf_close = gdf_pos.sort_values('distances')[1:n_closest]

        was_added_any = False
        for next_building_id, meters in zip(gdf_close['id'],
gdf_close['distances']):
            if next_building_id == building_id:
                continue

            graph.add_edge(building_id, next_building_id, meters=meters)
            was_added_any = True

        if not was_added_any:
            raise ValueError('...')

    print('The graph was build')

    add_stops(graph, pos_, stops_bus, 'bus')
    print('The bus were added')

    add_stops(graph, pos_, stops_tram, 'tram')
    print('The tram were added')

    add_stops(graph, pos_, stops_sub, 'subway')
    print('The subway were added')

    with open(cache_file_path, 'wb') as f:
        pickle.dump(graph, f)
    return graph

```

## ПРИЛОЖЕНИЕ 2

### Вторая часть кода для работы в графовой нейронной сети (обработка графа и вывод результатов)

```
import pickle
from pathlib import Path

import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.metrics import f1_score

import networkx as nx
import torch
from torch_geometric.data import Data, DataLoader
import torch.nn.functional as F
from torch_geometric.nn import GCNConv
filename = 'data_high.pkl'

def read_data() -> nx.Graph:
    cache_file_path = Path('data') / filename
    with open(cache_file_path, 'rb') as f:
        graph = pickle.load(f)

    return graph
G: nx.Graph = read_data()
N_FEATURES = 6

features = []
labels = []
node_mapping = {}
for i, (node_id, node_data) in enumerate(G.nodes(data=True)):
    node_mapping[node_id] = i
    is_stop = 0
    if node_data['label'] == 'bus':
        is_stop = 1
    elif node_data['label'] == 'tram':
        is_stop = 2
    elif node_data['label'] == 'subway':
        is_stop = 3
    features.append([
        node_data.get('square', 1),
        node_data['bar'] + node_data['pab'] + node_data['cafe'] +
node_data['fastfood'] + node_data['resto'] + node_data['bakery'],
        node_data['shoes'] + node_data['cloth'],
        node_data.get('hotel', 0),
        node_data['hair'] + node_data['beaty'],
        is_stop
    ])
    labels.append(int(node_data['is_flower'] > 0))
```

```

features = torch.tensor(features, dtype=torch.float)
num_nodes = features.size(0)

labels = torch.tensor(labels, dtype=torch.long)

train_indices, test_indices = train_test_split(
    np.arange(num_nodes),
    test_size=0.2,
    stratify=labels.numpy(),
    random_state=21
)

edges = list(map(lambda v: (node_mapping[v[0]], node_mapping[v[1]]),
G.edges))
edge_index = torch.tensor(edges, dtype=torch.long).t().contiguous()
data = Data(x=features, edge_index=edge_index, y=labels,
            train_mask=train_indices, test_mask=test_indices)

assert edge_index.max() < num_nodes, "Edge index is out of bounds!"

class GCN(torch.nn.Module):
    def __init__(self):
        super(GCN, self).__init__()
        self.conv1 = GCNConv(N_FEATURES, 32)
        self.bn1 = torch.nn.BatchNorm1d(32)
        self.conv2 = GCNConv(32, 64)
        self.bn2 = torch.nn.BatchNorm1d(64)
        self.conv3 = GCNConv(64, 128)
        self.bn3 = torch.nn.BatchNorm1d(128)

        self.conv5 = GCNConv(128, 2)

    def forward(self, x, edge_index):
        x = self.conv1(x, edge_index)
        x = self.bn1(x)
        x = F.relu(x)

        x = self.conv2(x, edge_index)
        x = self.bn2(x)
        x = F.relu(x)

        x = self.conv3(x, edge_index)
        x = self.bn3(x)
        x = F.relu(x)

        x = self.conv5(x, edge_index)
        return x

device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
print(device)
model = GCN().to(device)
data = data.to(device)

```

```

optimizer = torch.optim.Adam(model.parameters(), lr=0.0001)

class_weights = torch.tensor([1., 10.])
criterion = torch.nn.CrossEntropyLoss(weight=class_weights)

def calc_scores(out_, target_) -> tuple[float, float, float]:
    out_max = out_.max(1)[1]
    correct = target_.eq(out_max).sum().item()
    acc = correct / target_.size()[0]

    correct1 = target_[target_.eq(1)].eq(out_max[target_.eq(1)]).sum().item()
    acc1 = correct1 / target_[target_.eq(1)].size()[0]

    f1 = float(f1_score(target_.numpy(), out_max.numpy(), average='macro'))
    return f1, acc, acc1

model.train()
for epoch in range(1000):
    optimizer.zero_grad()

    out = model(data.x, data.edge_index)

    loss = criterion(out[data.train_mask], data.y[data.train_mask])
    loss.backward()

    optimizer.step()

    if epoch % 10 == 0:
        model.eval()
        out = model(data.x, data.edge_index)
        model.train()

        f1_train, acc_train, acc1_train = calc_scores(out[data.train_mask],
data.y[data.train_mask])
        f1_test, acc_test, acc1_test = calc_scores(out[data.test_mask],
data.y[data.test_mask])
        print(f'Epoch {epoch}, Loss: {loss.item():.4f}, '
              f'f1/train: {f1_train:.2f}, acc/avg/train: {acc_train:.2f},
acc/1/train: {acc1_train:.2f}, '
              f'f1/test: {f1_test:.2f}, acc/avg/test: {acc_test:.2f},
acc/1/test: {acc1_test:.2f}')

def save_to_qgis(data_, pred_):
    data_out = []
    for i, (node_id, node_data) in enumerate(G.nodes(data=True)):
        data_out.append([
            node_id,
            node_data['label'],

```



```

        node_data.get('square', 1),
        node_data['bar'],
        node_data['pab'],
        node_data['cafe'],
        node_data['fastfood'],
        node_data['resto'],
        node_data['bakery'],
        node_data['cloth'],
        node_data['shoes'],
        node_data['label'],
        node_data['is_flower'],
        node_data['coordinate'][1],
        node_data['coordinate'][0],
        pred[i]

    df_out = pd.DataFrame(data_out, columns=['building_id', 'label',
'square', 'bar', 'pab', 'cafe', 'fastfood', 'resto',
                                     'bakery', 'cloth', 'shoes',
'label', 'is_flower', 'lat', 'lon', 'predict'])

    df_out.iloc[data_.train_mask].to_csv(filename.replace('.csv',
'_train.csv'), index=False)
    df_out.iloc[data_.test_mask].to_csv(filename.replace('.csv',
'_test.csv'), index=False)

model.eval()
with torch.no_grad():
    pred = model(data.x, data.edge_index)
    f1, acc_test, acc1_test = calc_scores(pred[data.test_mask],
data.y[data.test_mask])

    save_to_qgis(data, pred.max(1)[1].numpy())

    print(f'TEST: f1: {f1:.2f}, acc/avg/test: {acc_test:.2f}, acc/1/test:
{acc1_test:.2f}')

```

## ПРИЛОЖЕНИЕ 3

### Код для модели классификации на примере *personal\_development* с процессом обучения модели и применения на новые данные

```
!pip install catboost lightgbm xgboost scikit-learn pandas
import pandas as pd
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.ensemble import RandomForestClassifier
from xgboost import XGBClassifier
from catboost import CatBoostClassifier
from lightgbm import LGBMClassifier
from sklearn.metrics import classification_report, f1_score, accuracy_score
import joblib

df = pd.read_csv('personal_development_edu.csv')
df_filtered = df[df['education'] == 1].copy()

if len(df_filtered) == 0:
    raise ValueError("Нет данных с education=1. Проверьте входные данные.")

X = df_filtered.drop(['personal_development', 'building_id', 'y', 'x'],
axis=1)
y = df_filtered['personal_development']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=21)

param_grids = {
    'Random Forest': {
        'model': RandomForestClassifier(random_state=42),
        'params': {
            'n_estimators': [50, 100, 200],
            'max_depth': [3, 5, 7, None],
            'min_samples_split': [2, 5],
            'class_weight': ['balanced', None]
        }
    },
    'XGBoost': {
        'model': XGBClassifier(random_state=42, use_label_encoder=False,
eval_metric='logloss'),
        'params': {
            'n_estimators': [50, 100, 200],
            'max_depth': [3, 5, 7],
            'learning_rate': [0.01, 0.1, 0.2],
            'scale_pos_weight': [1, (y_train == 0).sum()/(y_train ==
1).sum())]
        }
    },
    'LightGBM': {
        'model': LGBMClassifier(random_state=42),
        'params': {
            'n_estimators': [50, 100, 200],
```

```

        'max_depth': [3, 5, 7],
        'learning_rate': [0.01, 0.1],
        'class_weight': ['balanced', None]
    }
}

catboost_params = {
    'iterations': [50, 100],
    'depth': [3, 5, 7],
    'learning_rate': [0.01, 0.1],
    'auto_class_weights': ['Balanced', None]
}

results = {}

for name, config in param_grids.items():
    print(f"\nПодбор параметров для {name}...")
    grid_search = GridSearchCV(
        estimator=config['model'],
        param_grid=config['params'],
        cv=3,
        scoring='f1',
        n_jobs=-1,
        verbose=1
    )
    grid_search.fit(X_train, y_train)

    best_model = grid_search.best_estimator_
    y_pred = best_model.predict(X_test)

    joblib.dump(best_model, f'{name.lower().replace(" ",
    "_")}__best_model.pkl')

    results[name] = {
        'best_params': grid_search.best_params_,
        'f1_score': f1_score(y_test, y_pred),
        'accuracy': accuracy_score(y_test, y_pred),
        'report': classification_report(y_test, y_pred, zero_division=0)
    }

    print(f"Лучшие параметры {name}: {grid_search.best_params_}")
    print(f"F1-score: {results[name]['f1_score']:.4f}")
    print(f"Accuracy: {results[name]['accuracy']:.4f}")

print("\nПодбор параметров для CatBoost...")
cat_features = list(X.select_dtypes(include=['object', 'category']).columns)
grid_search_cb = GridSearchCV(
    estimator=CatBoostClassifier(
        random_state=42,
        cat_features=cat_features,

```

```

        verbose=0,
        eval_metric='F1'
    ),
    param_grid=catboost_params,
    cv=3,
    scoring='f1',
    n_jobs=-1,
    verbose=1
)
grid_search_cb.fit(X_train, y_train)

best_cb = grid_search_cb.best_estimator_
y_pred_cb = best_cb.predict(X_test)

joblib.dump(best_cb, 'catboost_best_model.pkl')

results['CatBoost'] = {
    'best_params': grid_search_cb.best_params_,
    'f1_score': f1_score(y_test, y_pred_cb),
    'accuracy': accuracy_score(y_test, y_pred_cb),
    'report': classification_report(y_test, y_pred_cb, zero_division=0)
}

print("\nРезультаты подбора параметров (оптимизация по F1-score):")
for name, res in results.items():
    print(f"\n{name}:")
    print(f"Лучшие параметры: {res['best_params']}")
    print(f"F1-score: {res['f1_score']:.4f}")
    print(f"Accuracy: {res['accuracy']:.4f}")
    print("Отчет классификации:\n", res['report'])

best_model_name = max(results, key=lambda x: results[x]['f1_score'])
best_model = joblib.load(f'{best_model_name.lower().replace(" ",
"_")}_best_model.pkl')
joblib.dump(best_model, 'best_overall_model.pkl')
print(f"\nЛучшая модель: {best_model_name} с F1-score
{results[best_model_name]['f1_score']:.4f}")

best_model = joblib.load('catboost_best_model.pkl')
X_all = df_filtered.drop(['personal development', 'building_id', 'y', 'x'],
axis=1)
predictions = best_model.predict(X_all)
probabilities = best_model.predict_proba(X_all)[:, 1]

results_df = df_filtered.copy()
results_df['predicted_class'] = predictions
results_df['prediction_probability'] = probabilities
results_df.to_csv('qgis_full_features_export.csv', index=False)
print("Результаты со всеми признаками сохранены в
qgis_full_features_export.csv")

```

```

import pandas as pd
import joblib

new_data = pd.read_csv('data_high14.csv')
new_data_filtered = new_data[new_data['education'] == 1].copy()
best_model = joblib.load('catboost_best_model.pkl')
cols_to_drop = ['personal_development', 'building_id', 'y', 'x']
X_new = new_data_filtered.drop([col for col in cols_to_drop if col in
new_data_filtered.columns], axis=1)
predictions = best_model.predict(X_new)
probabilities = best_model.predict_proba(X_new)[:, 1]

full_results = new_data.copy()
full_results['predicted_class'] = 0
full_results['prediction_probability'] = 0
full_results.loc[new_data['education'] == 1, 'predicted_class'] = predictions
full_results.loc[new_data['education'] == 1, 'prediction_probability'] =
probabilities

full_results.to_csv('full_predictions_with_all_data.csv', index=False)
print("Полные результаты сохранены в full_predictions_with_all_data.csv")

if {'y', 'x'}.issubset(full_results.columns):
    qgis_cols = ['building_id', 'y', 'x', 'predicted_class',
'prediction_probability'] + \
                [col for col in new_data.columns if col not in cols_to_drop]
    full_results[qgis_cols].to_csv('qgis_full_export.csv', index=False)
    print("Данные для QGIS сохранены в qgis_full_export.csv")

```