

УДК 004. 942+712.00  
doi: 10.15622/rcai.2025.094

## МАРКИРОВКА ВИДИМОСТИ ЭЛЕМЕНТОВ ВІМ-МОДЕЛЕЙ С ИСПОЛЬЗОВАНИЕМ ГРАФОВОЙ НЕЙРОННОЙ СЕТИ

Д.Д. Кабайлов (*denis2000kab@gmail.com*)<sup>A,B</sup>  
А.М. Наместников (*am.namestnikov@gmail.com*)<sup>A</sup>

<sup>A</sup> Санкт-Петербургский государственный  
архитектурно-строительный университет, Санкт-Петербург  
<sup>B</sup> ООО «Рэд Лайн Проект», Санкт-Петербург

В данной работе описан метод маркировки элементов цифровых информационных моделей зданий на определенных категориях видов с использованием графовой нейронной сети. В первом разделе работы приведен анализ существующих решений формирования чертежей их особенностей и недостатков. Во втором разделе описываются основные принципы построения и структура несвязного графа, представляющего совокупность информационных моделей зданий. Также приведены графические модели, отображающие полученный несвязный граф. В третьем разделе работы описаны причины выбора графовой нейронной сети Graph Attention Network (GAT) и слоев GATv2Conv. Описана структура нейросети, обучаемая на полученном графе, которая позволяет предсказывать метки видимости на видах для элементов информационных моделей зданий. Приведены результаты вычислительных экспериментов с разработанной нейросетевой моделью.

**Ключевые слова:** графовая нейронная сеть, сверточная нейронная сеть, GAT, ТИМ, проектирование.

### Введение

Создание чертежей зданий и сооружений является основополагающим этапом проектирования, в том числе в строительной сфере. Однако, чтобы создать качественный чертеж, необходимо определить оптимальное количество видов (планов/ фасадов/ разрезов и т.п.) и то, какие элементы на этих видах должны быть отображены.

В ряде исследований применялись алгоритмы классификации элементов информационных моделей, например в [Austern et al., 2024], однако они не позволяют определить, к каким видам какой элемент относится. Графовые нейросетевые модели Graph Convolutional Network и Graph Attention Network (GCN, GAT) [Çeter, 2024] демонстрируют перспективные результаты в смежных задачах аннотирования планов, но их точность и адаптивность пока недостаточны для масштабного применения. Таким образом, остаётся нерешённой задача автоматизированного формирования проектной документации, способной учитывать взаимосвязь всех элементов и атрибутов в модели.

Данные ограничения указывают на необходимость разработки моделей, которые будут учитывать геометрию и полный набор атрибутов элементов, обеспечивая корректное отображение на всех типах видов. Целью научного исследования является создание метода формирования проектной документации на основе интеграции технологий информационного моделирования (ТИМ) и нейросетевых алгоритмов анализа информационных моделей зданий и сооружений. Научная новизна работы состоит в исследовании способов адаптации и эффективности применения нейросетевых алгоритмов в задаче формирования чертежей зданий и сооружений.

Для достижения цели, принято решение провести ее декомпозицию на подзадачи. В рамках данной статьи приведен разбор следующих задач:

- анализ существующих подходов к формированию чертежей зданий и сооружений;
- описание метода анализа информационных моделей, использующего нейросетевой алгоритм, выполняющий маркировку видимости элементов на различных типах видов;
- анализ результатов исследования и разбор дальнейших шагов исследования.

В данной работе для классификации элементов по отображению на видах в модели принято решение использовать графовые нейросетевые алгоритмы, способные учитывать большой массив связанных данных. Это позволит создавать чертежные виды и выносить их на листы строительной документации.

## **1. Анализ современных подходов к формированию чертежей**

В настоящее время большинство задач в процессе формирования чертежей реализованы с помощью специально разработанных дополнительных надстроек для популярных систем автоматизированного проектирования (САПР). Системы на основе правил (rule-based) алгоритмов [Kim et al., 2024] позволяют формализовать отдельные этапы оформления чертежей, например простановку размеров и формирование спецификаций.

Однако такие решения, как показано в [Kim et al., 2024], не учитывают уже созданные виды и плохо масштабируются на проекты с различной структурой моделей.

В России достаточно много решений, имеющих инструменты по автоматизации формирования чертежей, одним из которых является плагин «ModPlus»<sup>1</sup> для ПО «Revit». Данные инструменты реализуют rule-based подход, ориентированный на дублирование и настройку видов, но не предоставляют возможность по созданию видов для информационных моделей с учетом всех атрибутов и связей элементов.

В работе [Çeter, 2024] автор представляет результаты исследования нейросетевых алгоритмов GCN и GAT, основанных на графах, в смежной задаче расстановки аннотированных обозначений на планах зданий: размеров, выносок, меток. В своей работе S. Çeter для создания информационных моделей использует программу ArchiCAD и сформированное приложение ServCAD. По итогам тренировки моделей автор выделяет следующие результаты:

- В задачах простановки марок для дверей и помещений наибольшую точность в размере 91% показала модель с архитектурой GAT.
- В задаче простановки размеров для стен обе модели показали достаточно низкую точность. Наибольшую точность в размере 41% показала модель с архитектурой GCN.

Автор указывает на высокую перспективность и репрезентативность использования графовых нейросетей в задачах, связанных с информационным моделированием.

В результате анализа можно сделать вывод, что большинство rule-based инструментов для решения конкретных задач процесса формирования чертежей представляют собой традиционные алгоритмы. Чем сложнее задача, тем специализированнее становится разрабатываемый алгоритм. Также существует алгоритм на основе машинного обучения, позволяющий проставлять аннотированные обозначения на планах зданий. Продемонстрирована достаточно успешная применимость графовых нейросетевых алгоритмов в смежных задачах ТИМ-проектирования [Çeter, 2024]. Однако работ, связанных с ТИМ и нейронными сетями, на текущий момент достаточно мало, что указывает на присутствие множества ниш и большое поле для исследований в области подготовки строительной документации с использованием ТИМ и нейросетевых алгоритмов.

---

<sup>1</sup> URL: <https://modplus.org/ru>.

## 2. Формализация задачи и представление информационных моделей зданий

Информационные модели зданий имеют явную графовую структуру, в которой:

- Узлы (vertices,  $V$ ) – элементы BIM-модели: стены, колонны, плиты.
- Рёбра (edges,  $E$ ) – связи между элементами: примыкание, расстояние и другие метрики, пригодные для описания взаимодействия элементов [Austern et. al., 2024].

Математически, каждая отдельная модель здания представляется графом:

где  $A$  – матрица признаков узлов,  $d$  – размерность векторного описания элемента.

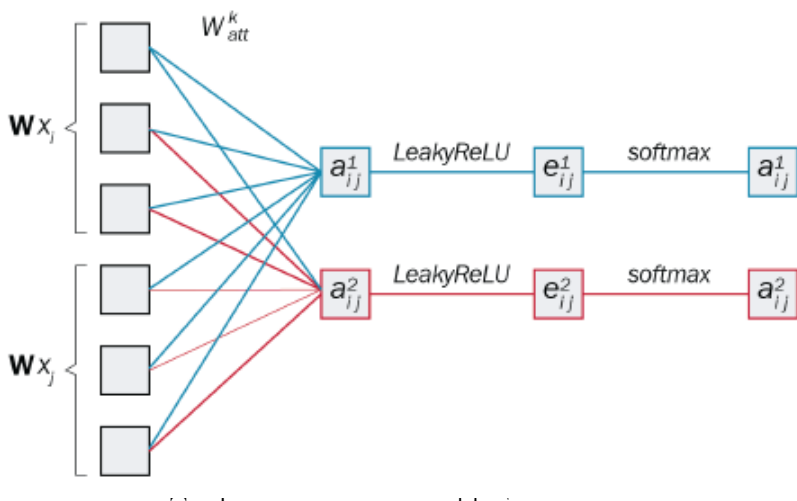
Особенность предлагаемого нейросетевого алгоритма заключается в необходимости его обучения на едином массиве данных. Формальное описание датасета будет представлять собой несвязный граф:

где  $M$  – количество отдельных информационных моделей, а индексы узлов дополнены префиксами для предотвращения пересечения идентификаторов.

### 2.1 Формальная модель нейронной сети

Для решения поставленной задачи архитектура нейросети должна включать методы учета влияния соседних узлов, чтобы лучше запоминать окружение и присваивать степень «важности», так как, например, для формирования плана здания степень «важности» между конструкциями, находящимися на одном уровне, выше, чем степень «важности» между конструкциями, расположенными на разных этажах [Лунева и др., 2024].

Чтобы реализовать учет «важности», необходимо применить технологию множественного внимания (multi-head attention). Данный способ применяется в нейросетевых алгоритмах вида GAT. Принцип работы данного алгоритма заключается в дополнительном использовании независимых «голов», формирующих индивидуальные векторы весов влияния от  $i$ -узла к  $j$ -узлу посредством объединения признаков (рис.). После вычисления весов, необходимо объединение полученных векторов, которое выполняется по двум схемам: вычисление вектора со средними значениями по всем векторам внимания и конкатенация всех векторов внимания [Labonne, 2023].



В результате формула для вычисления коэффициентов внимания имеет следующий вид [Labonne, 2023]:

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_k \exp(e_{ik})},$$

где имеет место:

- конкатенация признаков целевого узла и соседнего ;
- применение функции активации LeakyReLU;
- умножение на матрицу весов внимания ;
- softmax нормализация полученного результата.

Для множественного внимания с  $K$  «головами» формируется набор выходных признаков, которые содержат в себе данные не только об узле, но и о соседях и их влиянии:

где  $\sigma$  – функция активации, а  $\parallel$  обозначает конкатенацию по всем «головам».

Стоит отметить, что от выбора количества «голов» зависит качество признаков на выходе, так как каждая из них учитывает разные зависимости узлов.

## 2.2. Подготовка данных

В первой итерации в качестве обучающей и тестовой выборки были созданы 6 информационных моделей жилых многоэтажных зданий, созданных в программном обеспечении (ПО) «Revit». Для формирования датасета были созданы похожие модели с небольшими различиями: количество этажей, расположение и количество конструкций на этаже, расположение лестнично-лифтовых узлов, присутствие/отсутствие стен/колонн/балок. Данные по моделям сведены в таблицу.

Таблица 1

№ модели	Кол-во этажей	Кол-во элементов	№ модели	Кол-во этажей	Кол-во элементов
1	21	1751	4	20	1152
2	21	1822	5	27	1925
3	20	1107	6	27	2005

С помощью специально разработанного инструмента были выгружены необходимые для построения графов данные в виде трех текстовых файлов.

В первом файле содержатся массивы уникальных идентификаторов элементов с уже присвоенными префиксами и данными о геометрическом расположении (рис.):

где  $\mathbf{v}$  – вектор признаков узла,  $\mathbf{c}$  – категория узла,  $\mathbf{p}$  – координаты крайней левой нижней и крайней правой верхней точек описывающего элемент параллелепипеда.

```
[[
  [1751;-2000240;0;0;-9,84251968503937;0;0;-9,84251968503937],
  [24612;-2000240;0;0;0;0;0],
]]
```

Рис. 2. Пример выгруженных данных элементов

Во втором файле – массивы уникальных идентификаторов, между которыми необходимо построить ребра (рис.).

```
[[
  [509108;510268;510473;510588;510753;522444;522446;522448],
  [510268;510473;510753;515184;521567],
]]
```

Рис. 3. Пример выгруженных данных связей элементов

Третий файл датасета содержит элементы информационной модели (рис.). Такая структура обусловлена тем, что совокупность элементов позволяет определить расположение вида, а его категория – тип вида.

```
[[  
  [1752;1;509347;509461;509466;509471;509476],  
  [24631;2;509347;509466;509471;509476;509599;509709],  
  ]]
```

Рис. 4. Пример выгруженных данных видов

Реализация алгоритма анализа информационных моделей выполнена в среде разработки `jupyter notebook`<sup>1</sup> на языке `python` с использованием библиотеки `networkx`<sup>2</sup>. Для графического отображения графа была использована библиотека `matplotlib`<sup>3</sup>.

### 2.3. Формирование графовых представлений информационных моделей

Создание несвязного графа производится с использованием библиотеки `networkx`. В качестве входных параметров передается созданный экземпляр графа, данные о ребрах и узлах. Создание ребер выполняется по соответствующим данным, после чего для каждого полученного узла определяется вектор с данными об этом узле.

Графическое отображение полученного несвязного графа выведено при помощи функции `draw_networkx` модуля `networkx` (рис.).

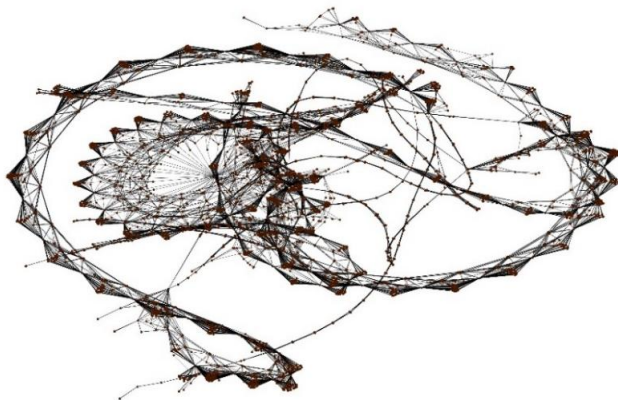


Рис. 5. Граф информационных моделей созданного датасета

<sup>1</sup> URL: <https://jupyter.org>.

<sup>2</sup> URL: <https://networkx.org>.

<sup>3</sup> URL: <https://matplotlib.org>.

На представленном графе отражены пять информационных моделей жилых многосекционных зданий. Общее количество узлов составляет 9762. Также в структуре графа необходимо отметить явную сходимость нескольких узлов в одном, что говорит о наличии общего элемента, например перекрытии. С целью подготовки данных графа для обучения моделей применялся модуль `torch_geometric`<sup>1</sup>.

### 3. Обучение нейросетевого алгоритма

#### 3.1. Подготовка графа для обучения нейросети

В представленной работе для создания нейросети используются библиотеки `torch`<sup>2</sup> и `torch_geometric`. Для того, чтобы произвести обучение моделей, предоставляемых этими пакетами, необходимо подготовить данные по созданному графу, так как по умолчанию они не совместимы. Далее граф преобразуется в формат библиотеки `torch_geometric` функцией:

К каждому узлу добавлены три дополнительных параметра: `degree`, `clustering`, `pagerank`. Параметр `degree` хранит в себе количество связей с текущим узлом, параметр `clustering` – показатель кластеризации данного узла, параметр `pagerank` – важность данного узла среди остальных. Также в функцию добавлено масштабирование признаков с использованием класса `StandardScaler`, как часто используемый [Akhatov et. al., 2024].

После преобразования данных производится разделение на обучающую и тестовые выборки с использованием масок. В первой итерации проводилось разделение по соотношению 0.8 ед. к 0.2 ед., где 0.8 ед. – обучающая выборка, а 0.2 ед. – тестовая выборка.

#### 3.2. Архитектура графовой нейросети

Разработанная нейронная сеть является сверточной с несколькими скрытыми слоями. Выбрана архитектура GAT, так как при помощи механизма внимания появляется возможность учитывать только важные соседние узлы, что необходимо при построении видов информационных моделей зданий. Диапазон скрытых слоев GATv2Conv выбран от 2 до 5. Также принято решение добавить функции нормализации данных и слои `dropout` с вероятностью 0.2 перед выходным слоем с целью предотвращения переобучения [Бабичева и др., 2021].

Критерием оптимизации выбран Adaptive Moment Estimation [Simon, 2024]. В качестве функции потерь была выбрана функция кросс-энтропии с весом 2, так как имеет место задача бинарной классификации.

---

<sup>1</sup> URL: <https://pytorch-geometric.readthedocs.io/en/latest/index.html>.

<sup>2</sup> URL: <https://pytorch.org>.



Стоит отметить, что в работе для каждого типа вида создается своя модель машинного обучения, так как для построения определенного вида необходимо учитывать узлы по разным критериям. Например, чтобы построить план, необходимо учесть расположение всех конструкций на этаже, а для фрагмента достаточно учитывать локальные зоны, но также важно проанализировать расстояние между конструкциями.

### 3.3. Обучение нейросети и анализ результатов

На первом этапе обучения количество перцептронов в скрытом слое было установлено в размере 64. Было установлено количество эпох обучения, равное 30. Также проведено обучение моделей с разным количеством скрытых слоев в диапазоне от 2 до 5 и разными массивами «голов». В результате получены графики потерь и точности для всех моделей, а также в работе представлены графики для моделей, предсказывающей планы и фрагменты планов, с 5 скрытыми слоями и для модели, предсказывающей разрезы, с 2 скрытыми слоями (рис. 6-8). Результаты тестирования точности моделей сведены в таблицу.

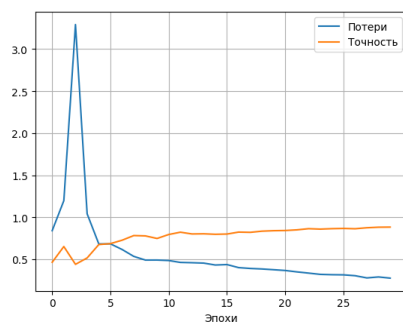


Рис. 6. Графики потерь и точности модели, предсказывающей планы

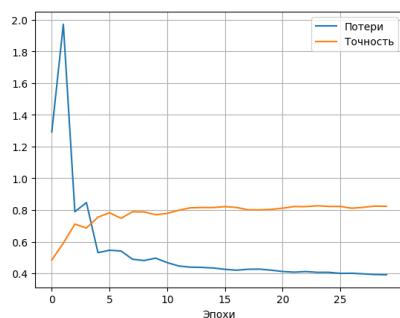


Рис. 7. Графики потерь и точности модели, предсказывающей разрезы

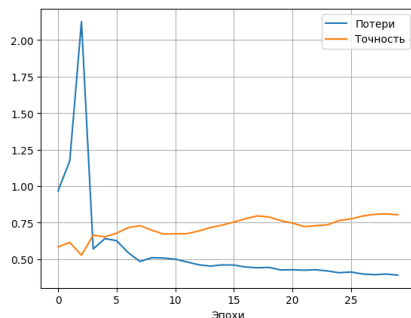


Рис. 8. Графики потерь и точности модели, предсказывающей фрагменты планов

По графикам модели планов заметно, что начиная с 20 эпохи точность растет меньше и близка к 0.8, но ошибка продолжает существенно уменьшаться.

По графикам модели разрезов заметно, что, начиная с эпохи 10, точность и потери стабилизируются. Такое поведение связано с тем, что в моделях мало разрезов, поэтому модель запоминает только конкретные паттерны.

По графикам модели фрагментов планов заметно, что модель на протяжении всего обучения имеет высокую амплитуду точности. Это связано с тем, что модель не может найти нужный паттерн и в данном случае необходимо провести дополнительный анализ архитектуры нейросети.

По результатам, приведенным в таблица 2, можно сделать следующие выводы:

1. Для модели, предсказывающей разрезы на данный момент достаточно 2 скрытых слоя нейронной сети.
2. Для моделей, предсказывающей планы и фрагменты плана, увеличение количества скрытых слоев приводит к увеличению точности.

Таблица 2

Тип вида	2 слоя	3 слоя	4 слоя	5 слоев
План	82.96%	79.81%	83.32%	87.05%
Разрез	79.59%	79.96%	80.83%	79.66%
Фрагмент плана	75.13%	81.49%	81.20%	83.32%

## Заключение

В данной работе выполнен анализ существующих подходов и решений по формированию чертежей зданий. Предложен и реализован метод классификации элементов цифровых информационных моделей зданий по признаку видимости на различных типах видов, основанный на использо-

вании графовых нейронных сетей. Для проведения вычислительных экспериментов создан датасет в виде несвязного графа, содержащий множество подграфов – информационных моделей. Также в работе была реализована подготовка данных, включая извлечение топологических признаков (degree, clustering, pagerank) и нормализацию признаков.

Разработаны, обучены и протестированы модели графовых нейронных сетей с разным количеством слоев GATv2Conv от 2 до 5. Наилучшие результаты (87.05% и 83.32% соответственно) для планов и фрагментов планов показали модели с пятью скрытыми слоями. Для модели, предсказывающей разрезы на данный момент достаточно двух скрытых слоя.

Выявлены ограничения, связанные с недостатком объема данных для некоторых типов видов (фасады, фрагменты-сечения), а также нестабильностью обучения модели для фрагментов планов, что свидетельствует о необходимости расширения датасета и адаптации архитектуры под каждый отдельный тип вида. Также необходимо провести более детальное тестирование применяемых функций потерь и алгоритмов оптимизации.

### Список литературы

- [Austern et al., 2024] Austern G., Bloch T., Abulafia Y. Incorporating Context into BIM-Derived Data—Leveraging Graph Neural Networks for Building Element Classification // Buildings. – 2024. – Vol. 14(2). – P. 527. – DOI: <https://doi.org/10.3390/buildings14020527>.
- [Çeter, 2024] Server Çeter Automated BIM Model Annotation via Graph Neural Networks: Bridging the Gap Between Design and Documentation: Master Thesis for the Master of Science Program in Civil Engineering / TUM School of Engineering and Design, 2024. – P. 89.
- [Kim et al., 2024] Kim Y., Chin S., Choo S. Rule-based automation algorithm for generating 2D deliverables from BIM // Journal of Building Engineering. – 2024. – Vol. 97. – P. 111033. – DOI: <https://doi.org/10.1016/j.jobbe.2024.111033>.
- [Лунева и др., 2024] Лунева Е.Е., Банокин П.И. Графовые нейронные сети в задачах продленной аутентификации групповых профилей социальных сетей // System Analysis and Mathematical Modeling. – 2024. – Т. 6, № 3. – С. 300-311. – DOI 10.17150/2713-1734.
- [Labonne, 2023] Maxime Labonne Hands-On Graph Neural Networks Using Python: Livery Place, 35 Livery Street, Birmingham, B3 2PB, UK, Packt Publishing, 2023.
- [Akhatov et al., 2024] Akhatov A.R., Renavikar A., Rashidov A.E.O., Nazarov F.M. Optimization of the number of databases in the big data processing// Проблемы информатики. – 2023. – No. 1(58). – P. 33-47. – DOI 10.24412/2073-0667-2023-1-33-47.
- [Бабичева и др., 2021] Бабичева М.В., Данилов В.В., Борщевский С.В. Вероятностный подход к оптимизации нейронных сетей случайной редукцией нейронов // Вестник Донецкого национального университета. Серия Г: Технические науки. – 2021. – № 1. – С. 63-70.
- [Simon, 2024] Simon J.D. Prince Understanding Deep Learning. 255 Main Street, 9th Floor, Cambridge, MA 02142 MIT Press, 2024.