

ФОРМУВАННЯ НАБОРУ ДАНИХ BIM-МОДЕЛІ ДЛЯ ЗАДАЧ МАШИННОГО НАВЧАННЯ

Вінницький національний технічний університет

У статті представлено відтворюваний підхід до формування наборів даних із BIM-моделі Autodesk Revit для подальшого застосування методів машинного навчання у проектуванні. Запропоновано чотири взаємодоповнювальні маршрути експорту (Schedules→CSV, Dynamo до CSV/XLSX/JSON, pyRevit/Revit API до CSV/JSON/XLSX, IFC до APS/Speckl до JSON/CSV) та уніфікований «договір даних»: цільову схему/онтологію з ядром обов'язкових полів, SI-одинацями, стабільною політикою ключів (IfcGUID/композитний ключ) і формальною валідацією якості. Методологія охоплює інгестію даних, нормалізацію типів і одиниць, формування дескрипторів геометрії, перевірку повноти й референтної цілісності, а також підготовку ознак для надійного поділу train/val/test.

Показано приклад практичного протоколу та кінцевого датасету Elements.csv, який використовується для базового прогнозу маси елементів як проксі вартості й логістичних обмежень; наведено блок-схеми конвеєрів і шаблони перевірок якості. На прикладних задачах (оцінка темпів робіт, орієнтовна вартість, оптимізація компонування та несучих конструкцій) продемонстровано, що дисципліна даних (схема, ключі, одиниці, правила) знижує ризики витоку інформації та підвищує відтворюваність висновків.

Практичний внесок роботи полягає в узгодженні інструментальних маршрутів із єдиною схемою представлення, що дає змогу інтегрувати BIM-дані з аналітичними процесами без залежності від конкретного ПЗ й створює основу для даноцентричного проектування та подальшої оптимізації. Окремо описано артефакти відтворюваності (схеми JSON/Table Schema, кваліметричні звіти DQI, журнали перетворень), що полегшують аудит даних і повторення експериментів.

Ключові слова: BIM (Інформаційне моделювання в будівництві), Revit, оптимізація конструкцій Dynamo, Python, Машинне навчання, MCE.

Вступ

Створення надійного та відтворюваного набору даних BIM для завдань оптимізації структурного планування є нетривіальним завданням через фрагментовані джерела, різноманітні параметри та залежність від власних середовищ моделювання. Геометричні, матеріальні та контекстні атрибути елементів типової BIM-моделі розміщуються в різних просторах і з різними одиницями виміру, що ускладнює їх комбінування, контроль якості та відстеження між версіями елементів. Це виправдовує стандартизований експорт у табличні та напівструктуровані представлення (CSV, XLSX, JSON), що гарантує взаємодію, доступ до машин та незалежність від конкретного програмного забезпечення.

Правильна структура є важливою умовою: ідентифікація цільової схеми/онтології з необхідними полями, узгодженими типами та одиницями SI, стабільною політикою ключів (IfcGUID/композитний ключ) та формалізованою перевіркою якості (перевірка діапазону, пропуски, дублікати, референтна цілісність). При підготовці даних для машинного навчання така організація безпосередньо впливає на валідність ознак, контроль витоку інформації та відтворюваність експерименту (тренування/валідація/тестування, фіксація початкових даних). Зокрема, при оптимізації розташування конструкцій (балки, колони, перекриття, фундаменти) високоякісні дані дозволяють оцінити вплив геометрії, матеріалу та просторового контексту на такі критерії, як вартість, вага, жорсткість, обмеження монтажу та зіткнення.

Метою статті обґрунтувати потребу експорту даних в табличні варіанти, описати принципи структуризації та продемонструвати, як перетворені дані стають придатними для алгоритмів машинного навчання.

Постановка проблеми

У задачах оптимізації розташування конструктивних елементів будівлі (балок, колон, плит, фундаментів, ферм, арматури) ключовим обмеженням є відсутність надійного, відтворюваного та машинно-видимого датасету, сформованого безпосередньо з BIM-моделі. Наявні BIM-сховища орієнтовані на візуально-геометричну репрезентацію та інтерактивну взаємодію, тоді як алгоритми машинного навчання потребують стабільних ідентифікаторів, уніфікованих одиниць вимірювання, чітких типів даних і контрольованої семантики полів. Розрив між моделюванням і аналітикою призводить до труднощів з трасованістю даних, неконтрольованого витоку інформації, неможливості порівнювати результати між версіями моделі та до зниження валідності висновків ML.

Дані в BIM розпорошені між рівнями (Levels), типами/екземплярами (Type/Instance), категоріями та сімействами, зберігаються у різних одиницях і охоплюють як табличні, так і геометрично-ієрархічні структури. Експорт “як є” породжує неоднорідність, пропуски, дублікати, несталі ключі (наприклад, зміни UniqueId при копіюванні) і втрати контексту (фаза, системна належність, кімнатно-поверховий зв’язок). Без спеціальної цільової схеми/онтології ці недоліки накопичуються й руйнують відтворюваність ML-експериментів.

Основний матеріал

У сучасних робочих процесах BIM дані більше не є побічним продуктом моделювання, а натомість є основним активом управління життєвим циклом об'єкта. Експорт із середовища BIM у відкриті/напіввідкриті формати (CSV, JSON, XLSX) вирішує три основні проблеми:

1. Гарантує сумісність інструментів (CAD, PLM, ERP, CDE, APS/Speckle тощо);
2. Прокладає шлях до аналітики та машинного навчання без залежності від власних API;
3. Розширює відтворюваність та відстежуваність інженерних рішень шляхом керування версіями та регульованого перетворення одиниць вимірювання.

Для завдань оптимізації розташування конструкцій (балок, колон, плит, фундаментів) експорт допомагає чітко пов'язати геометричні, матеріальні та контекстні параметри з цільовими параметрами (вартість, маса, жорсткість, обмеження монтажу), підготувати елементи, створити опорні моделі та регулярно оцінювати вплив факторів.

Експорт має підкорятися цільовій схемі/онтології, що визначає склад і семантику полів, їх типи та одиниці. Рекомендовані принципи:

- *Ядро ідентифікації*: DocumentGUID, ModelVersion, ExportTimestampUTC, ElementUniqueId, за можливості - IfcGUID; для відстеження у часі - композитний ключ K, який поєднує у собі декілька параметрів ідентифікації і являється об'єднаним параметром;
- *Таксономія*: Category, Family, Type, InstanceOrType, Level, Phase, System/Discipline;
- *Метрики в SI*: довжини (м), площі (м²), об'єми (м³), кути (° або рад – вибрати одне), з фіксованою точністю та документованими перетвореннями з внутрішніх одиниць;
- *Геометричні репрезентації*: для табличної аналітики - скаляри (наприклад, Area_m2, Volume_m3, CentroidX/Y/Z_m, Orientation); складну геометрію (меші, криві) - або агрегувати до дескрипторів, або зберігати окремо у JSON;
- *Типізація й кодування*: числа - десяткова крапка; логічні - true/false; дати/час - ISO 8601 (UTC); кодування - UTF-8; категоріальні поля - із контрольованими доменами (ENUM/довідники);
- *Якість і валідація*: схеми рівня файлу (JSON Schema / Table Schema), перевірки діапазонів, пропусків, унікальності ключів, референтної цілісності; автоматичний DQI-звіт.

Формати в які експортуються дані з BIM моделі можуть відрізнятися в залежності від їх використання. Основні варіанти даних будуть зберігатися у таких форматах як (CSV, XLSX, JSON)

CSV – основний носій для табличних фактів (мільйони рядків, мінімальний оверхед). Один файл – один тип сутності (наприклад, Elements.csv), без масивів у комірках. Перевага: сумісність з більшістю інструментів даних і ML.

JSON – для ієрархічних або рідкісних параметрів, вкладених структур (матеріали елемента, списки армування, геометричні властивості). Рекомендовано супроводжувати JSON Schema для перевірки

XLSX – зручний для ручного огляду/рев'ю, контрольних вивантажень, комунікації з нефаховими користувачами; не бажаний як “production” формат сховища.

Таким чином, експортовані дані - це контракт між BIM і аналітикою: вони повинні бути формально описані, стабільні в ідентифікації, уніфіковані в одиницях, перевірені на якість і структуровані з урахуванням подальшого використання в алгоритмах оптимізації та машинного навчання. Це перетворює BIM на джерело надійного знання, придатного для прийняття обґрунтованих інженерних рішень.

Базовий граф експорту задається чотирма маршрутами, кожен з яких має власний профіль переваг та обмежень. Перший маршрут - експорт специфікацій Revit (Schedules) безпосередньо у CSV - мінімізує залежності від сторонніх бібліотек і забезпечує керовану вибірку полів та фільтрів. Другий - Dynamo (версія 2024) - надає low-code/visual-logic доступ до параметрів і дозволяє писати

у CSV, XLSX і JSON (через вузли або скрипт-вузол Python), що зручно для пакетних вивантажень по кількох категоріях. Третій - pyRevit / Revit API - надає повний контроль над ітерацією елементів, нормалізацією одиниць (перехід до SI), побудовою геометричних дескрипторів і серіалізацією у цільові формати з явною типізацією. Нарешті, четвертий - маршрут IFC→веб-API (Autodesk Platform Services) або сервер даних Speckle - відкриває міжплатформний доступ до властивостей елементів як JSON із використанням IfcGUID як стабільного ідентифікатора. Для наглядності наведемо схеми кожного варіанта:

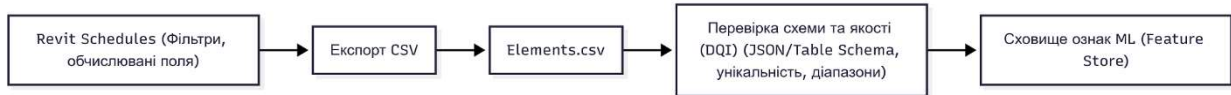


Рисунок 1 – Специфікації Revit → CSV

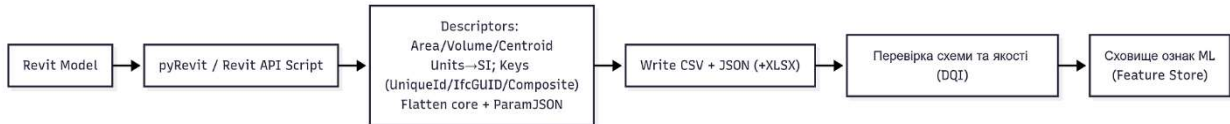


Рисунок 2 – Dynamo → CSV/XLSX/JSON

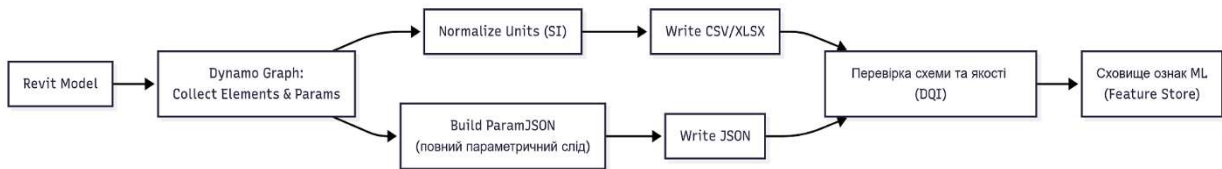


Рисунок 3 – pyRevit / Revit API → CSV/JSON/XLSX

Незалежно від маршруту експорту, методологія конвеєра незмінна:

- Визначення області експорту (конструктивні категорії),
- Вилучення параметрів і побудова геометричних дескрипторів,
- Нормалізація одиниць до SI та явна типізація (числа, булеві, дати/час ISO 8601 UTC, категоріальні домени),
- Приведення даних до “tidy”-подання та/або «зіркової схеми» (фактова таблиця з довідниками),
- Валідація якості за формальною схемою (обов'язкові поля, діапазони, унікальність ключів, референтна цілісність, outliers),
- Підготовка до ML (кодування категорій, масштабування числових, фіксований seed для сплітів train/val/test).

Щоб конкретизувати, наведено міні-фрагмент кінцевого датасету. Припустимо, що Revit-2024, ціль - конструктивні категорії, нормалізація SI, часовий пояс - UTC.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	DocumentGUID,ModelVersion,ExportTimestampUTC,ElementUniqueId,IfcGUID,Category,Family,Type,InstanceOrType,Level,Phase,Area_m2,Volume_m3,Thickness_m,CentroidX_m,CentroidY_m,CentroidZ_m,PrimaryMaterial,IsDeleted														
2	3a1ce5f,v1,2025-08-26T12:00:00Z,1b2ce5f,,Structural Framing,UB,UB 457x152x52,Instance,Level 2,New,0.285,12.40,8.10,6.20,S355,false														
3	3a1ce5f,v1,2025-08-26T12:00:00Z,9f7ee5f,,Floors,RC Slab,250 mm,Type,Level 1,New,32.540,8.135,0.250,14.05,5.20,3.15,Concrete C30/37,false														
4	3a1ce5f,v1,2025-08-26T12:00:00Z,7a99e5f,,Structural Columns,UC,UC 254x254x89,Instance,Level 1,New,8.120,2.030,10.22,4.33,4.10,S355,false														
5															

Рисунок 4 – Приклад кінцевого файлу CSV

В даному файлі описані назви параметри елементів та їх значення. В даному файлі використані такі параметри як:

- DocumentGUID – Номер документа, в якому знаходиться модель;
- ModelVersion – Номер останньої збереженої версії документу;
- ExportTimestampUTC – Дата експорту даних елемента;
- ElementUniqueId – Унікальний ID в межах файлу;
- IfcGUID – Унікальний код для IFC моделі;
- Category – Категорія сімейства Revit;
- Family – Назва сімейства;

- Type – Назва типу;
- InstanceOrType – Елемент типу чи Екземпляру сімейства;
- Level – Рівень до якого прикріплений елемент;
- Phase – Стадія зведення;
- Area_m2 – Площа елементу;
- Volume_m3 – Об'єм елементу;
- Thickness_m – Товщина;
- CentroidX_m – Центр BoundaryBox по осі X;
- CentroidY_m – Центр BoundaryBox по осі Y;
- CentroidZ_m – Центр BoundaryBox по осі Z;
- PrimaryMaterial – Основний матеріал;
- IsDeleted – Чи був елемент видалений, чи існує досі;

Формальна сторона супроводжується простими, але дисциплінованими схемами контролю: для CSV - одна сутність на файл, без масивів у комірках; десяткова крапка, UTF-8, явна одиничність ключів; для JSON - валідація через JSON Schema (Draft 2020-12) з required, type, minimum/maximum, enum для категоріальних полів. Таке подвійне представлення - "плоске ядро" + ParamJSON - гарантує одночасно сумісність із табличними ML-алгоритмами та збереження повного параметричного сліду для аудиту/розширених методів.

Такий мінімум демонструє, що правильно експортовані й структуровані дані з BIM - із фіксованою політикою ключів (IfcGUID/компонентний), SI-одинацями, tidy-поданням, формальною валідацією та серіалізацією в CSV/JSON/XLSX - стають відтворюваною основою для подальшого інтелектуального аналізу. Поєднання чотирьох маршрутів експорту з єдиною цільовою схемою/онтологією унаочнює ключовий принцип: незалежно від технологічної труби, «договір даних» залишається незмінним—стислий факт-шар для моделей і комплексний параметричний слід для аудиту, перевірок якості та розширюваності досліджень. Це і є практичний шлях перетворення BIM-моделі на надійний датасет для оптимізації розташування конструкцій та інших ML-задач у дослідницьких і прикладних сценаріях.

Готовий датасет Elements.csv є фактовою таблицею рівня екземпляра конструктивних елементів із нормалізованими SI-одинацями, стабільними ключами (ElementUniqueId, IfcGUID, компонентний ключ) та базовими геометричними/матеріальними дескрипторами. Його подальше використання у ML вимагає:

1. Формальної перевірки схеми та якості,
2. Агрегування й фічірінгу під конкретну постановку (регресія/класифікація),
3. Контрольованого протоколу розбиття на train/val/test для запобігання витоку інформації,
4. Системи контролю версій/дрейфу.

Практичне використання починається із «приймання» даних. Ви завантажуєте новий «Elements.csv» і запускаєте перевірку відповідності паспортів схем: контроль наявності обов'язкових полів, правильності одиниць, допустимих діапазонів (наприклад, площа й об'єм не можуть бути від'ємними), відсутності повторів за ключем і дотримання зв'язків із довідниками типів, матеріалів, рівнів.

Результат такої перевірки доцільно оформляти коротким звітом якості: скільки записів, скільки порожніх значень у кожному полі, які саме правила порушені, чи є підозрілі викиди (надто великі або малі значення). Якщо порушення є, їх спершу усувають (виправлення одиниць, перейменування полів за таблицею відповідності, дозаповнення пропусків із довідників), і лише після цього набір допускається до аналітики.

Коли дані узгоджені й збагатилися необхідними показниками, формується робоча таблиця для розрахунків. Якщо завдання стосується окремих елементів, зазвичай достатньо «плоского» подання: один рядок – один елемент, відібрані числові та текстові поля, чисті одиниці, без порожніх значень у критичних стовпцях. Якщо завдання визначене на рівні груп (поверх, система, тип), дані спершу агрегують: підраховують суму площ або об'ємів, середні значення відміток, кількість елементів певного виду на рівень тощо. Агрегація має бути прозорою: чітко вказано, за якою ознакою групували та які саме показники обчислювали.

Особливу увагу слід приділяти поділу на вибірки для побудови й перевірки моделей. Щоб не «підглянути» майбутні результати, поділ роблять так, аби близькі за змістом об'єкти не опинилися одночасно і в навчальній, і в контрольній частинах. Для будівельних даних зручно ділити за рівнями, за документами або за типами. Це забезпечує чесну перевірку: модель бачить у навчанні одні групи, а перевіряється на інших, подібних, але не тих самих.

Експорт із BIM у структуровані таблиці (на кшталт Elements.csv) перетворює модель на джерело вимірюваних величин: геометрію (площі, об'єми, товщини), контекст (рівні, фази, системи), матеріали та інші параметри. Далі формується послідовність кроків:

- Перевірка якості (повнота, унікальність ключів, одиниці SI, допустимі межі);
- Збагачення (приєднання довідників: щільність матеріалів, норми продуктивності, прайси);
- Виведення показників (маса = об'єм × щільність; фронт робіт на рівень тощо);
- Моделювання (оцінка або прогноз цільової величини) та
- Прийняття рішення (вибір сценарію, який задовольняє обмеження й мінімізує/максимізує обраний критерій).

Це дозволяє переходити від «переліку елементів» до керованих рішень у плануванні, кошторисі та конструктиві.

Одні з таких керованих рішень це:

- Прогнозування темпів будівництва (виробничі графіки);
- Прогнозування вартості будівництва (кошторисні оцінки);
- Оптимізація схеми (компонування та послідовність);
- Оптимізація несучих конструкцій;
- Контроль ризиків.

Щоб результати були надійні, застосовується подвійний контроль в який входить: **Порівняння з еталонами**. Вибірка відомих ділянок (із вимірними темпами/вартістю) для оцінки похибки;

Аналіз чутливості. Визначення, які параметри найсильніше впливають на результат (наприклад, зміна товщини плити на 10 % як позначається на масі та тривалості).

Прогнозні значення й ознаки, що впливають на рішення, повертаються до середовища спільної роботи як мітки або допоміжні параметри (наприклад, «очікувана тривалість», «вартість на елемент», «пріоритет монтажу»). Завдяки сталій системі ідентифікаторів (IfcGUID/композитний ключ) ці дані точно прив'язуються до вихідних елементів, що забезпечує відслідковуваність та можливість повторної перевірки.

Висновки

У статті показано, що перетворення BIM-моделі (Revit 2024) у структуровані дані (CSV/JSON/XLSX) є необхідною передумовою для відтвореної аналітики та застосування методів машинного навчання у проєктуванні. Запропоновано чотири взаємодоповнювальні маршрути експорту - через специфікації (Schedules), Dynamo, pyRevit/Revit API та IFC→APS/Speckle - які, попри різну складність, зводяться до єдиного “договору даних”: цільової схеми з ядром обов'язкових полів, уніфікованими SI-одиницями, стабільною політикою ключів (IfcGUID/композитний ключ) та формальною валідацією якості.

Показано, як така схема забезпечує трасованість елементів між версіями, контроль відсутності дублікатів і коректність типів, а також як перетворювати геометричні та контекстні параметри на узгоджені дескриптори. Окреслено повний конвеєр даних: від інгестії та збагачення довідниками до підготовки ознак, чесного поділу на train/val/test і базового моделювання. На прикладних задачах (прогноз темпів і вартості, оптимізація схеми та несучих конструкцій) продемонстровано, що дисципліна даних безпосередньо підвищує якість рішень і зменшує ризики.

Практичний внесок роботи - уніфікований протокол експорту, шаблони перевірки якості, приклади коду (low-code і code-first) та наочні блок-схеми. Подальші дослідження варто спрямувати на розширення довідників матеріалів і норм виробітку, автоматизоване виявлення дрейфу даних, а також інтеграцію з репозитаріями ознак для багатооб'єктних порівнянь. Сукупно це формує надійну основу для даноцентричного проєктування й оптимізації у сучасному BIM.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Кишкан А., Кархут, І. (2024). ПОТЕНЦІАЛ АВТОМАТИЗАЦІЇ ТА ПРИСКОРЕННЯ БУДІВЕЛЬНИХ ПРОЦЕСІВ ЗА ДОПОМОГОЮ МАШИНОГО НАВЧАННЯ ТА ШТУЧНОГО ІНТЕЛЕКТУ. Матеріали конференцій МЦНД, (17.05.2024; Ужгород, Україна), 354–358. вилучено із <https://archive.mcmd.org.ua/index.php/conference-proceeding/article/view/1231>
2. Сьомко, П., & Левус, Є. (2024). ВИКОРИСТАННЯ МАШИНОГО НАВЧАННЯ ДЛЯ ВИЗНАЧЕННЯ АРХІТЕКТУРНОГО СТИЛЮ БУДІВЕЛЬ. Матеріали конференцій МНЛ, (17 травня 2024 р., м. Київ), 346–347. Вилучено з <https://archive.liga.science/index.php/conference-proceedings/article/view/964>

3. Адаменко В. Досвід впровадження BIM-технологій в навчальний процес на кафедрі металевих і дерев'яних конструкцій КНУБА. Будівельні конструкції. Теорія і практика. 2022. (онлайн-публікація). URL: <https://bctp.knuba.edu.ua/article/view/260026>
4. Ясній В. П., Мещерякова О. М. BIM: ефективний інструмент для реконструкції будівель та споруд. Сучасні технології та методи розрахунків у будівництві. 2022. Вип. 18. С. 61–70. DOI: [https://doi.org/10.36910/6775-2410-6208-2022-8\(18\)-08](https://doi.org/10.36910/6775-2410-6208-2022-8(18)-08)
5. Лялюк, О. Г., Осипенко, Р. С., & Мельник, Д. О. (2024). СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ В БУДІВЕЛЬНИХ ПРОЕКТАХ НА ОСНОВІ МЕРЕЖ БАЙЄСА. Сучасні технології, матеріали і конструкції в будівництві, 36(1), 96–102. <https://doi.org/10.31649/2311-1429-2024-1-96-102>
6. Лялюк, О. Г., & Осипенко, Р. С. (2023). ОСОБЛИВОСТІ ІМПЛЕМЕНТАЦІЇ ШТУЧНОГО ІНТЕЛЕКТУ В БУДІВНИЦТВІ. Сучасні технології, матеріали і конструкції в будівництві, 35(2), 172–176. <https://doi.org/10.31649/2311-1429-2023-2-172-176>
7. Назіров Е. К. Методи збору та класифікації даних за допомогою саундлетної байєсівської нейронної мережі / Е. К. Назіров, Т. О. Назірова, М. Ю. Карпенко // Вісник Херсонського національного технічного університету. – 2018. – № 3(1). – С. 332-337. – Режим доступу: [http://nbuv.gov.ua/UJRN/Vkhdtu_2018_3\(1\)_49](http://nbuv.gov.ua/UJRN/Vkhdtu_2018_3(1)_49).
8. buildingSMART International. IFC 4.3 Formally Approved and Published as an ISO Standard (ISO 16739). 02.04.2024. URL: <https://www.buildingsmart.org/ifc-4-3-formally-approved-and-published-as-an-iso-standard/>
9. Потенціал автоматизації та прискорення будівельних процесів за рахунок машинного навчання та штучного інтелекту / О. А. Бойко, А. С. Ковальчук // Modern Construction and Architecture. – 2023. – № 3. – С. 85–92. – Режим доступу: <https://archive.mcnd.org.ua/index.php/conference-proceeding/article/view/1231>
10. Autodesk Help. Export a Schedule (Revit 2024). 2024. URL: <https://help.autodesk.com/view/RVTLT/2024/ENU/?guid=GUID-B2CCAC4F-1D38-4D5D-B4D1-95619D1B7EBE>
11. Поняття про генетичні алгоритми та їх застосування в задачах оптимізації / О. В. Кравченко, П. І. Тимошенко // Комп'ютерні технології обробки даних. – 2023. – № 10. – Режим доступу: <https://jktod.donnu.edu.ua/article/view/16228>
12. Zheng, H.; Moosavi, V.; Akbarzadeh, M. Machine Learning Assisted Evaluations in Structural Design and Construction. Automation in Construction 2020, 114, 103346. DOI: <https://doi.org/10.1016/j.autcon.2020.103346>
13. scikit-learn. User Guide (stable): Model selection & evaluation. 2024–2025. URL: https://scikit-learn.org/stable/user_guide.html
14. Boiko, N.I.; Blazhevskiy, S.H. Method of Determining the Structure of the Model of Optimal Complexity. Herald of Khmelnytskyi National University, 2022, 2(307), 7–13. DOI: <https://doi.org/10.31891/2307-5732-2022-307-2-7-13>
15. Pyrih, Y.; Klymash, M.; Pyrih, Y.; Lavriv, O. Genetic Algorithm as a Tool for Solving Optimisation Problems. Information and Communication Technologies, Electronic Engineering 2023, 3(2), 95–107. DOI: <https://doi.org/10.23939/ictee2023.02.095>
16. Stetsyuk, P.I.; Vakulenko, D.V.; Stetsyuk, M.H. r-Algorithm for Learning Linear Regression Models. Problems of Applied Mathematics and Mathematical Modeling 2023, 23,. DOI: <https://doi.org/10.15421/322324>
17. Rochefort-Beaudoin, T.; Vadean, A.; Aage, N.; Achiche, S. Structural Design Through Reinforcement Learning. arXiv 2024, arXiv:2407.07288. DOI: <https://doi.org/10.48550/arXiv.2407.07288>
18. Kumar, A.; Singh, R. Application of Artificial Intelligence in Structural Engineering. Proceedings of the 2023 ASCE International Conference on Computing in Civil Engineering, 2023, pp. 350–359. DOI: <https://doi.org/10.1061/9780784485231.044>
19. Sadeghi Eshkevari, S.; Sadeghi Eshkevari, S.; Sen, D.; Pakzad, S.N. RL-Controller: A Reinforcement Learning Framework for Active Structural Control. arXiv 2021, arXiv:2103.07616. DOI: <https://doi.org/10.48550/arXiv.2103.07616>
20. Zhang, Y.; Wang, L.; Wang, Y.; Liu, Y.; Zhang, J. A Review of Artificial Intelligence Applications in Structural Engineering. Computers in Industry 2022, 142, 103750. DOI: <https://doi.org/10.1016/j.compind.2022.103750>

REFERENCES

1. Kyshkan, A., & Karkhut, I. (2024). Potential for Automation and Acceleration of Construction Processes Using Machine Learning and Artificial Intelligence. Conference Proceedings of the International Scientific Center of Development (Uzhhorod, Ukraine, 17 May 2024), 354–358. URL: <https://archive.mcnd.org.ua/index.php/conference-proceeding/article/view/1231>
2. Syomko, P., & Levus, Ye. (2024). Using Machine Learning to Determine the Architectural Style of Buildings. Conference Proceedings of MNL (Kyiv, 17 May 2024), 346–347. URL: <https://archive.liga.science/index.php/conference-proceedings/article/view/964>
3. Adamenko, V. (2022). Experience of Implementing BIM Technologies in the Educational Process at the Department of Metal and Timber Structures of KNUCA. Building Constructions. Theory and Practice. (Online publication). URL: <https://bctp.knuba.edu.ua/article/view/260026>

4. Yasnii, V. P., & Meshcheriakova, O. M. (2022). BIM: An Effective Tool for the Reconstruction of Buildings and Structures. *Modern Technologies and Methods of Calculations in Construction*, 18, 61–70. [https://doi.org/10.36910/6775-2410-6208-2022-8\(18\)-08](https://doi.org/10.36910/6775-2410-6208-2022-8(18)-08)
5. Lialiuk, O. H., Osypenko, R. S., & Melnyk, D. O. (2024). Decision Support Systems in Construction Projects Based on Bayesian Networks. *Modern Technologies, Materials and Structures in Construction*, 36(1), 96–102. DOI: <https://doi.org/10.31649/2311-1429-2024-1-96-102>
6. Lialiuk, O. H., & Osypenko, R. S. (2023). Features of the Implementation of Artificial Intelligence in Construction. *Modern Technologies, Materials and Structures in Construction*, 35(2), 172–176. DOI: <https://doi.org/10.31649/2311-1429-2023-2-172-176>
7. Nazirov, E. K., Nazirova, T. O., & Karpenko, M. Yu. (2018). Methods of Data Collection and Classification Using a Soundlet Bayesian Neural Network. *Bulletin of the Kherson National Technical University*, 3(1), 332–337. URL: [http://nbuv.gov.ua/UJRN/Vkhdtu_2018_3\(1\)_49](http://nbuv.gov.ua/UJRN/Vkhdtu_2018_3(1)_49)
8. buildingSMART International. (2024). IFC 4.3 Formally Approved and Published as an ISO Standard (ISO 16739). URL: <https://www.buildingsmart.org/ifc-4-3-formally-approved-and-published-as-an-iso-standard/>
9. Boiko, O. A., & Kovalchuk, A. S. (2023). The Potential of Automation and Acceleration of Construction Processes through Machine Learning and Artificial Intelligence. *Modern Construction and Architecture*, 3, 85–92. URL: <https://archive.mend.org.ua/index.php/conference-proceeding/article/view/1231>
10. Autodesk Help. (2024). Export a Schedule (Revit 2024). URL: <https://help.autodesk.com/view/RVTLT/2024/ENU/?guid=GUID-B2CCAC4F-1D38-4D5D-B4D1-95619D1B7EBE>
11. Kravchenko, O. V., & Tymoshenko, P. I. (2023). Concepts of Genetic Algorithms and Their Application to Optimization Problems. *Computer Technologies of Data Processing*, URL: <https://jktod.donnu.edu.ua/article/view/16228>
12. Zheng, H., Moosavi, V., & Akbarzadeh, M. (2020). Machine Learning Assisted Evaluations in Structural Design and Construction. *Automation in Construction*, 114, 103346. URL: <https://doi.org/10.1016/j.autcon.2020.103346>
13. scikit-learn. (2024–2025). User Guide (stable): Model Selection & Evaluation. URL: https://scikit-learn.org/stable/user_guide.html
14. Boiko, N. I., & Blazhevskiy, S. H. (2022). Method of Determining the Structure of the Model of Optimal Complexity. *Herald of Khmelnytskyi National University*, 2(307), 7–13. DOI: <https://doi.org/10.31891/2307-5732-2022-307-2-7-13>
15. Pyrih, Y., Klymash, M., Pyrih, Y., & Lavriv, O. (2023). Genetic Algorithm as a Tool for Solving Optimisation Problems. *Information and Communication Technologies, Electronic Engineering*, 3(2), 95–107. DOI: <https://doi.org/10.23939/ictee2023.02.095>
16. Stetsyuk, P. I., Vakulenko, D. V., & Stetsyuk, M. H. (2023). r-Algorithm for Learning Linear Regression Models. *Problems of Applied Mathematics and Mathematical Modeling*, 23. DOI: <https://doi.org/10.15421/322324>
17. Rochefort-Beaudoin, T., Vadean, A., Aage, N., & Achiche, S. (2024). Structural Design Through Reinforcement Learning. *arXiv*, arXiv:2407.07288. DOI: <https://doi.org/10.48550/arXiv.2407.07288>
18. Kumar, A., & Singh, R. (2023). Application of Artificial Intelligence in Structural Engineering. In *Proceedings of the 2023 ASCE International Conference on Computing in Civil Engineering* (pp. 350–359). DOI: <https://doi.org/10.1061/9780784485231.044>
19. Sadeghi Eshkevari, S., Sadeghi Eshkevari, S., Sen, D., & Pakzad, S. N. (2021). RL-Controller: A Reinforcement Learning Framework for Active Structural Control. *arXiv*, arXiv:2103.07616. DOI: <https://doi.org/10.48550/arXiv.2103.07616>
20. Zhang, Y., Wang, L., Wang, Y., Liu, Y., & Zhang, J. (2022). A Review of Artificial Intelligence Applications in Structural Engineering. *Computers in Industry*, 142, 103750. DOI: <https://doi.org/10.1016/j.compind.2022.103750>

Андрухов Валерій Михайлович – к.т.н., доцент, Вінницький національний технічний університет, e-mail: vmandruhov@gmail.com. ORCID: 0000-0002-4749-8569

Потеха Андрій Сергійович – аспірант 1 курсу, факультет будівництва, цивільної та екологічної інженерії Вінницький національний технічний університет, м. Вінниця. e-mail: apotexa67@gmail.com. ORCID: 0009-0008-8187-1049

V. M. Andrukhov
A. S. Potiekha

FORMATION OF BIM MODEL DATASET FOR MACHINE LEARNING TASKS

Vinnytsia National Technical University;

This paper presents a reproducible approach to deriving machine-learning-ready datasets from Autodesk Revit Building Information Models. We unify four complementary export routes—Schedules→CSV, Dynamo→CSV/XLSX/JSON, pyRevit/Revit API→CSV/JSON/XLSX, and IFC→Autodesk Platform Services/Speckle→JSON/CSV—under a single data contract: a target schema/ontology with mandatory fields, SI units, a stable key policy (IfcGUID/composite key), and formal

data-quality validation. The methodology covers ingestion, normalization of types and units, construction of geometric descriptors, checks of completeness and referential integrity, and feature preparation for reliable train/validation/test splits.

Provided a practical protocol and a final Elements.csv dataset used in a baseline task that predicts element mass as a proxy for cost and logistics; templates for quality checks accompany the flow. Across applied scenarios—production-rate estimation, cost approximation, and optimization of layout and load-bearing systems—we show that disciplined data management (schema, keys, units, rules) reduces information leakage and improves reproducibility.

The contribution aligns tool-specific export routes with a uniform representation, enabling integration of BIM data with analytics without dependence on a software stack and establishing a foundation for data-centric design and subsequent optimization. We also document reproducibility artefacts (JSON/Table Schema definitions, DQI reports, and processing logs) to support audit and repetition.

Keywords: *BIM, Revit, optimization of constructions, Dynamo, Python, Machine Learning, FEA.*

Andrukhov Valeriy– PhD, Associate Professor, Vinnytsia National Technical University, e-mail: vmandruchov@gmail.com. ORCID: 0000-0002-4749-8569

Potiekha Andriy– student, Department of Civil and Environmental Engineering Vinnytsia National Technical University, Vinnytsia city. e-mail: apotexa67@gmail.com. ORCID: 0009-0008-8187-1049