

**МЕТОД ВИЯВЛЕННЯ ДУБЛІКАТІВ WORKFLOW
ДЛЯ LORA-МОДЕЛЕЙ НА ОСНОВІ КАНОНІЗАЦІЇ ГРАФОВИХ
СТРУКТУР ТА СТРУКТУРНОГО ХЕШУВАННЯ**

Лі Н.Д.

nikita.li@nure.ua

Науковий керівник – канд. техн. наук, ст. викл. Кобилін І.О.
Харківський національний університет радіоелектроніки, каф. ІНФ
м. Харків, Україна

This paper proposes a method for detecting duplicate and near-duplicate workflows used in LoRA-based image generation systems. Workflows exported in JSON format are modeled as directed graphs and transformed into canonical representations independent of node identifiers and interface metadata. The proposed approach combines structural normalization, topological sorting, cryptographic hashing (SHA-256) and SimHash-based similarity analysis. The method enables accurate detection of both exact duplicates and slightly modified workflow versions. Experimental evaluation demonstrates high precision and low false-positive rates in a simulated marketplace environment. The developed solution can be integrated into AI workflow marketplaces to ensure content uniqueness and protect intellectual property.

З розвитком генеративних моделей штучного інтелекту, зокрема LoRA-моделей для генерації зображень, активно розвиваються маркетплейси workflow для систем типу Stable Diffusion та ComfyUI. Workflow у таких системах являє собою структуровану конфігурацію обчислювального процесу, яка визначає послідовність операцій, параметри моделей, способи підключення додаткових модулів та взаємодію між окремими компонентами генеративної архітектури. Фактично workflow виступає формалізованим описом алгоритму створення зображення з урахуванням конкретних налаштувань семплерів, LoRA-адаптацій, масштабування, постобробки та інших операцій.

Активне поширення таких конфігурацій призвело до формування спеціалізованих маркетплейсів, де користувачі можуть публікувати, продавати або обмінюватися готовими workflow. Проте однією з ключових проблем функціонування подібних платформ є повторне розміщення однакових або незначно модифікованих workflow. Зловмисники або недобросовісні користувачі можуть змінювати ідентифікатори вузлів, порядок елементів у JSON-файлі або незначні параметри, зберігаючи при цьому фактичну структуру обчислювального графа. Це призводить до дублювання цифрового контенту, зниження довіри до платформи, порушення принципів інтелектуальної власності та ускладнення модерації маркетплейсу.

Відсутність ефективних механізмів автоматичної перевірки унікальності workflow створює додаткові ризики для авторів оригінальних

конфігурацій та негативно впливає на екосистему генеративних AI-платформ загалом. Саме тому розробка формалізованого методу виявлення точних і структурно подібних дублікатів є актуальним науково-прикладним завданням.

Workflow експортується у форматі JSON та представляється як орієнтований граф $G = (V, E)$, де V — множина вузлів, E — множина зв'язків між ними. Кожен вузол містить тип операції та набір параметрів.

Формально процес канонізації можна подати у вигляді: $C(G) = \text{TopSort}(\text{Normalize}(G))$ де $\text{Normalize}(G)$ — процедура видалення службових атрибутів та впорядкування ключів, TopSort — топологічне сортування графа, $C(G)$ — канонічне представлення workflow. Криптографічний хеш визначається як: $H(G) = \text{SHA256}(C(G))$

Запропонований метод складається з трьох етапів. На першому етапі виконується нормалізація структури шляхом видалення службових полів (ідентифікаторів вузлів, координат інтерфейсу, часових міток) та впорядкування JSON-ключів. На другому етапі здійснюється канонізація графа через топологічне сортування, що дозволяє отримати унікальне представлення незалежно від порядку збереження вузлів.

Для точного виявлення дублікатів обчислюється криптографічний хеш $H(G) = \text{SHA256}(C(G))$, де $C(G)$ — канонічне представлення графа. Для виявлення псевдодублікатів використовується SimHash та косинусна міра подібності між векторами ознак workflow.

Workflow вважається потенційним псевдодублікатом, якщо значення подібності перевищує порогове значення 0,9. Експериментальне моделювання на наборі зі 100 workflow показало 100% виявлення точних копій та понад 90% виявлення структурно подібних версій.

Приклад нормалізації:
Початковий фрагмент JSON:

```
{ "id": 12,  
  "type": "LoRALoader",  
  "x": 124,  
  "y": 88 }
```

Після нормалізації:

```
{  
  "type": "LoRALoader"  
}
```

Після канонізації структура графа може набувати вигляду:
[CheckpointLoader → LoRALoader → KSampler → SaveImage]

Таким чином, різні представлення одного й того самого workflow зводяться до єдиного канонічного вигляду.

Запропонований метод може бути інтегрований у маркетплейси AI-контенту для автоматичного контролю унікальності публікацій та забезпечення чесної конкуренції між авторами.

Алгоритм 1. Виявлення дублікатів workflow

Input: JSON-файл workflow W

Output: Ознака exact або near-duplicate

1. Видалити службові поля з W
2. Відсортувати ключі JSON
3. Побудувати граф $G = (V, E)$
4. Виконати топологічне сортування
5. Отримати канонічне представлення $C(G)$
6. Обчислити $H = \text{SHA256}(C(G))$
7. Якщо H існує у базі $\rightarrow$ exact duplicate
8. Обчислити $\text{SimHash}(G)$
9. Якщо $\text{similarity} > 0.9 \rightarrow$ near-duplicate

Список використаних джерел:

1. Ding S.X. Model-Based Fault Diagnosis Techniques. Springer, 2009.
2. Mallat S., Zhang Z. Matching pursuits with time-frequency dictionaries. IEEE Transactions on Signal Processing, 1993.
3. Thapa C. et al. SplitFed: When Federated Learning Meets Split Learning. AAAI, 2022.
4. Vepakomma P. et al. Split learning for health. arXiv, 2018.
5. Li Z. et al. Adaptive Split Learning over Energy-Constrained Wireless Edge Networks. arXiv, 2024.