

УДК 004.42:004.738.5:519.72

DOI <https://doi.org/10.32782/2415-8151.2025.35.26>

СТВОРЕННЯ ІНТЕРАКТИВНИХ ВІЗУАЛІЗАЦІЙ У ВЕБЗАСТОСУНКАХ

Зозуля Юлія Олександрівна

аспірантка кафедри дизайну і технологій,
Київський національний університет культури і мистецтв, Київ, Україна,
e-mail: julja.sotnikova@gmail.com, orcid: 0009-0008-4075-0267

Анотація. Мета. Метою статті є розроблення методів інтерактивної візуалізації даних в інтерфейсах вебзастосунків на базі силових орієнтовних графів. У статті представлено процес розроблення та впровадження інтерактивної візуалізації даних у вебзастосунках, зосереджено увагу на силових орієнтовних графах як засобі динамічного представлення і аналізу складних наборів даних.

Методологія. Основним підходом стало впровадження патерну «вхід – оновлення – вихід», що забезпечує динамічне зв'язування даних з елементами DOM для створення, оновлення та видалення візуальних елементів відповідно до змін у наборі даних. Прикладним методом реалізовано силові орієнтовні графи, де поведінка вузлів моделюється за фізичними принципами. Ураховано сили притягування та відштовхування, які описуються законами Гука та Кулона, для підтримання оптимальних відстаней між вузлами, а також застосовано позиційні сили для утримання графів у межах області перегляду.

Результати. Було визначено, що силові орієнтовані графи, що складаються з кількох непересічних підграфів, моделюються з використанням фізичних принципів, які керують поведінкою їхніх елементів. Ці візуалізації врівноважують сили відштовхування та притягування, щоб зберегти просторову чіткість і уникнути накладання вузлів, водночас застосовують позиційні коригування, щоб гарантувати, що всі підграфи залишаються в єдиній формі. У статті доведено необхідність інтеграції бібліотеки D3.js, яка полегшує пов'язування даних з об'єктною моделлю документа (DOM). Ця інтеграція базується на шаблоні «вхід – оновлення – вихід», що дозволяє динамічно створювати, змінювати та видаляти візуальні елементи для відображення нових наборів даних. Шаблон гарантує, що кожен елемент даних унікально представлений і постійно оновлюється у відповідь на зміни, досягаючи надійного узгодження даних із візуалізацією.

Наукова новизна полягає в тому, що було визначено порядок застосування передових методів кластеризації для вдосконаленої організації даних. Високимірні вкладання, отримані з моделей обробки природної мови, було використано для захоплення семантичних і контекстних зв'язків у наборах даних. Ці вкладання додатково оптимізовано за допомогою методів зменшення розмірності, як-от t-SNE та UMAP, які зберігають значущі зв'язки, водночас спрощують структуру даних. Доведено, що отримані кластери необхідно представляти візуально, щоб підкреслити семантичну зв'язність, покращити інтерпретацію та допомогти у виявленні прихованих шаблонів.

Практична значущість. Щоб продемонструвати практичну корисність запропонованої методології, її було застосовано до набору даних електронної комерції із семінару електронної комерції "SIGIR 2021". Цей набір даних охоплює мільйони взаємодій і обширні метадані, забезпечує реальний контекст для

оцінювання ефективності візуалізації. Цей підхід дає змогу кластеризувати продукти на основі спільних характеристик, полегшує розуміння поведінки споживачів і їхнього ставлення до продуктів.

Ключові слова: інтерактивна репрезентація, силово-орієнтовані графи, зв'язування даних, інтеграція d3.js, методи кластеризації, редукція розмірності.

ВСТУП

У розвитку інформаційних технологій, спрямованих на розроблення програмного забезпечення, останнім часом спостерігається тенденція більш об'ємного охоплення функціональних даних у вебзастосунках. Дана тенденція насамперед пов'язана зі сталим удосконаленням процесів розроблення програмного забезпечення, що так чи інакше впливає на його масштабування. Незважаючи на фактичну користь таких трансформаційних змін, із цим виникає проблема оптимізованої візуалізації даних у програмних графічних інтерфейсах вебзастосунків. Центральний корінь даної проблеми полягає у браку більшої інтерактивності. Менша інтерактивність є причиною погіршеної інтерпретації відображених даних, може спричинити каскад наступних небажаних наслідків, що загалом може цілком компрометувати бекенд-складник вебзастосунку.

Отже, поточний підхід фокусуватиметься на імплементації патерну «вхід – оновлення – вихід» в архітектурі D3.js для створення інтерактивної візуалізації даних, шляхом зв'язування даних, специфікуючись на анімованих кулькових діаграмах як прикладному дослідженні.

АНАЛІЗ ПОПЕРЕДНІХ ДОСЛІДЖЕНЬ

У сучасних наукових колах досить активно розвивається напрям, що зосереджується на створенні новітніх методів для інтерактивної візуалізації даних, особливо в контексті розроблення інтерфейсів програмного забезпечення. Це стало особливо актуальним через зростання потреб у більш зручних і ефективних способах представлення великих обсягів даних у цифрових продуктах, де важлива роль відводиться наочності й інтуїтивності взаємодії користувача з інформацією. На тлі цих змін з'являються нові наукові праці, які досліджують підходи й інструменти для реалізації інтерактивних візуалізацій, що завдяки привнесенню естетичних елементів здатні покращити сприйняття складних даних і забезпечити більш ефективне їх використання.

Окресленою науковою проблематикою цікавились переважно зарубіжні

дослідники, а саме: Т. Баркер, Д. Вестфол [1], С. Чонг, Ю. Сі [2], Р. Вонг [3], О. Ембарак [4], М. Гунатілака, Н. Перера, П. Вірасінгха [5], М. Іглесіас [6], Д. Кадієвич [7], Дж. Кімунге [8], Г. Кумар [9], Л. Куо, Т. Чанг, Лай С.-С. [10], К. Лі, Л. Ні, Л. Бай [11], Дж. Лу [12], Д. Матейзель [13], М. Морєб [14], Е. Морт, С. Брукнер, Н. Сміт [15], С. Нелс, Дж. Люде [16], А. Сух, М. Хаджідж, Б. Ван, С. Шайдеггер, П. Розен [17], С. Ван, Ю. Фань, Х. Лі, Ю. Лю [18], К. Чжан [19], Ф. Чжун, М. Сюе, Чжан Дж., Ф. Чжан, Р. Бан, О. Деуссен, Ю. Ван [20] та інші.

А. Сух, М. Хаджідж, Б. Ван, К. Шайдеггер, П. Розен [17] запропонували підхід до вдосконалення макетів силових орієнтованих графів шляхом використання персистивної гомології (persistent homology (далі – PH)) для вирішення проблем візуалізації складних графів. PH, інструмент топологічного аналізу даних, використовується для ідентифікації та маніпулювання ключовими структурними особливостями графа на основі їхнього значення стійкості. Описується виділення 0-вимірних характеристик PH як зі зважених, так і з незважених неорієнтованих графів, що дозволяє ідентифікувати структурні характеристики без необхідності вибору параметрів. Ці функції ранжуються за значущістю та візуалізуються за допомогою постійних штрихкодів, які служать інтерактивним інтерфейсом для зміни макетів графів.

Запропонований метод уводить дві керовані користувачем сили: звуження, яке зближує вузли, пов'язані з функціями з низьким рівнем стійкості, і відштовхування, яке розділяє вузли в окремих розділах графа. За допомогою цих сил користувачі можуть підкреслювати конкретні структури графів, водночас зменшувати візуальний безлад. Методологія демонструється на синтетичних і реальних наборах даних, демонструє її здатність покращувати читабельність графів і виділяти важливі закономірності [17].

За допомогою кількісних і якісних оцінок показано, що цей підхід дозволяє створювати макети, які краще відображають вибрані користувачем функції графа порівняно зі звичайними методами, спрямованими на силу та на основі кластеризації. Результати також підкреслюють масштабованість методу для

більших графів. Також була обговорена його інтеграція з існуючими інструментами візуалізації, обмеження, пов'язані з надмірно обмеженими макетами, і потенційні розширення функцій РН з більшою розмірністю для майбутніх досліджень.

М. Гунатілака, Н. Перера, П. Вірасінгхе [5] розробили автоматизовану систему для перетворення символів з паперових морських карт (paper nautical charts (далі – PNC)) на їхні еквіваленти електронних навігаційних карт (electronic navigational chart (далі – ENC)) з використанням моделі згорткової нейронної мережі на основі YOLOv5. Мотивація випливає з неефективності та схильності до помилок природи поточного ручного процесу розпізнавання та перетворення символів.

Методологія передбачала збір і попередню обробку даних символів зі сканованих паперових діаграм для створення надійного набору даних для навчання моделі виявлення об'єктів на основі YOLOv5. Анотація символів була виконана за допомогою обмежувальних рамок для точного позначення символів діаграми, що дозволило точне навчання моделі. Дослідження оцінювало різні варіації моделі YOLOv5, причому YOLOv5m продемонстрував високу продуктивність, досягнув середньої точності 0,837. Ефективність системи було підтверджено за допомогою експериментів із використанням різних показників ефективності, зокрема й матриць точності, запам'ятовування та перплексивності. Хоча модель продемонструвала високу точність у визначенні загальних символів, як-от «Перешкода» та «Головні / поміжні вогні», вона зіткнулася із проблемами в розрізненні менш помітних символів, як-от «Видимі уламки» [5].

Автоматизована система інтегрує графічний інтерфейс користувача для простоти використання, усуває потребу в детальних знаннях програмування. Ця інновація зменшує кількість людських помилок, прискорює створення карт і підвищує точність баз даних ENC відповідно до стандартів Міжнародної гідрографічної організації. Незважаючи на такі обмеження, як дисбаланс класів у наборі даних і випадкова неправильна класифікація текстових елементів як символів, дослідження забезпечує базову структуру для розпізнавання символів діаграм, керованих штучним інтелектом [5].

Проте, беручи до уваги вищезазначену наукову документацію із проблеми розроблення методів інтерактивної візуалізації даних, в інтерфейсах усе ще залишається значна кількість невисвітлених аспектів, що потребують подальшого опрацювання.

МЕТА

Мета – розроблення методів інтерактивної візуалізації даних в інтерфейсах вебзастосунків на базі силових орієнтовних графів.

РЕЗУЛЬТАТИ ТА ОБГОВОРЕННЯ

Силові орієнтовні графи складаються з декількох не зв'язаних один з одним підграфів у межах однієї візуалізації. Поведінкові механізми елементів таких графів моделюються, базуючись на фізичних принципах. Симуляція типово включає обчислення рівнодійної сили, яка діє на кожний вузол, залежно від його взаємодій з іншими вузлами та сполученнями. Головні сили використовують принципи відштовхування, аналогічно до закону Кулона, що запобігають перекриттю вузлів, взаємовідштовхують їх:

$$F_r = \frac{q_i q_j}{k_r d_{ij}^2}, \quad (1)$$

де i, j позначають вузли, k_r – постійна репрезентація сили відштовхування, q_i та q_j – заряди відповідних вузлів, d_{ij} – відстань між вузлами.

З іншого боку, сили притягування, на основі описаних у законі Гука, наближають зв'язані один з одним вузли ближче один до одного, витримують оптимальний проміжок, який запобігає ущільненню графів у кластер:

$$F_a = k_a (d_{ij} - d_0), \quad (2)$$

де k_a відповідає динамічній константі, яка визначає жорсткість сполучення, d_0 – бажана залишкова відстань сполучення

Головна проблема полягає у складності підтримання всіх підграфів у межах однієї області перегляду, оскільки стандартні центрувальні сили часто не здатні перешкодити зміщенню окремих підграфів за межі області перегляду. Отже, позиційні сили застосовуються з кінцевою метою долання зазначених труднощів за посередництвом форсованого підштовхування вузлів за спрямування до специфікованих позицій уздовж осей x та y , результуючи у відсутність суворого вирівнювання по центру:

$$F_x = k_x x_i; \quad (3)$$

$$F_y = k_y y_i, \quad (4)$$

де k_x та k_y є константами, які регулюють інтенсивність описаних сил.

Зв'язування даних у контексті D3.js полягає в асоціюванні елементів даних з елементами об'єктної моделі документів (Document Object Model (далі – DOM)). Таке зв'язування передбачає динамічне створення, оновлення та видалення візуальних елементів у відповідь на мінливий пул даних. Патерн «вхід – оновлення – вихід» є основою цього підходу й уможливорює наявність візуальної репрезентації в кожній частині даних і її подальше відповідне оновлення.

Вхідна ланка обробляє елементи даних, які не мають відповідних DOM-елементів. Далі нові елементи створюються та додаються до DOM-елементів для репрезентації цих даних. Ланка оновлення керує наявними елементами даних, за умови відповідності наявним DOM-елементам. Візуалізація підлаштовується до оновлень властивостей цих елементів для остаточного віддзеркалювання будь-яких змін у пулі даних. Вихідна ланка

відповідає за менеджмент навантаженням, спричиненого наявними DOM-елементами, які вже не мають відповідних даних у даних мережі.

Припускається, що $D = \{d_1, d_2, \dots, d_n\}$ позначає набір даних, а набору DOM-елементів відповідає $E = \{e_1, e_2, \dots, e_m\}$. У цьому відношенні метою є встановлення бієктивної функції $f: D \rightarrow E$, щоб кожна точка даних d_i присвоювалася унікальному DOM-елементу e_j . За умови, що набір даних змінюється до D , описаний патерн сприяє підлаштуванню E таким чином, щоб нова функція $f': D' \rightarrow E'$ точно репрезентувала оновлені дані.

Створення інтерактивного неперетинаючого силового графа починається зі встановлення HTML-середовища та включення бібліотеки D3.js у нього. Елемент Scalable Vector Graphics (далі – SVG) слугує візуалізаційною структурою. Це описується у фрагменті коду в табл. 1.

Таблиця 1

Установлення HTML-середовища та включення D3.js-бібліотеки

```
<head>
<meta charset="UTF-8">
<title>Disjoint Force-Directed Graph</title>
<script src="https://d3js.org/d3.v6.min.js"></script>
<style>
body {
font-family: Arial, sans-serif;
}
svg {
border: 1px solid #ccc;
}
.link {
stroke: #999;
stroke-opacity: 0.6;
}
.node circle {
stroke: #fff;
stroke-width: 1.5px;
}
</style>
</head>
<body>
<svg id="graph" width="800" height="600"></svg>
<script>
</script>
</body>
</html>
```

Симуляція конфігурується таким чином, щоб розпізнавати та реагувати на елементи, до яких прив'язуються дані, з подальшим пристосовуванням сил і позицій. Зміст табл. 2 ілюструє даний процес у середовищі D3.js.

Відповідно до того, як набір даних змінюється – чи то за додавання нових вузлів, чи то оновлення наявних, або видалення їх, – досліджуваний патерн ураховує ці зміни, що продемонстровано в табл. 3.

З урахуванням даної функції, сполучення разом з вузлами перезв'язуються з оновленими даними. Нові елементи додаються до нових вхідних даних, наявні елементи оновлюються для відображення змін, а застарілі елементи піддаються видаленню. Далі дана симуляція перезапускається з метою акомодатії до нової структури даних, що забезпечує динамічне оновлення графа відповідно до розвитку пулу даних.

Таблиця 2

Конфігурація симуляції силового графа в середовищі D3.js

```

const simulation = d3.forceSimulation()
  .force("link", d3.forceLink().id(d => d.id).distance(50))
  .force("charge", d3.forceManyBody().strength(-100))
  .force("x", d3.forceX().strength(0.1))
  .force("y", d3.forceY().strength(0.1));

const link = svg.append("g")
  .attr("class", "links")
  .selectAll("line")
  .data(data.links)
  .enter().append("line")
  .attr("stroke-width", d => Math.sqrt(d.value));

const node = svg.append("g")
  .attr("class", "nodes")
  .selectAll("circle")
  .data(data.nodes)
  .enter().append("circle")
  .attr("r", 5)
  .attr("fill", d => color(d.group))
  .call(d3.drag()
    .on("start", dragstarted)
    .on("drag", dragged)
    .on("end", dragended));

```

Таблиця 3

Оновлення елементів графа у відповідь на зміни в наборі даних

```

function updateGraph(newData) {
  // Update links
  const linkUpdate = svg.select(".links")
    .selectAll("line")
    .data(newData.links, d => `${d.source.id}-${d.target.id}`);
  linkUpdate.enter().append("line")
    .attr("stroke-width", d => Math.sqrt(d.value));
  linkUpdate.exit().remove();
  const nodeUpdate = svg.select(".nodes")
    .selectAll("circle")
    .data(newData.nodes, d => d.id);
  nodeUpdate.enter().append("circle")
    .attr("r", 5)
    .attr("fill", d => color(d.group))
    .call(d3.drag()
      .on("start", dragstarted)
      .on("drag", dragged)
      .on("end", dragended));
  nodeUpdate.exit().remove();
  simulation.nodes(newData.nodes);
  simulation.force("link").links(newData.links);
  simulation.alpha(1).restart();
}

```

Прикладне дослідження ефективності запропонованої методології зосереджується на імплементації її фундаментальних принципів у галузі електронної комерції. Набір даних, застосований у цьому дослідженні, походить із конкурсу даних SIGIR 2021 E-Commerce Workshop та відображає багатогранні взаємодії в межах середньомасштабної електрокомерційної вебплатформи "Shop Z". Цей набір даних охоплює понад 10 мільйонів інтеракцій із продуктами та 57 483 унікальні одиниці складського обліку (Stock Keeping Unit (далі – SKU)). Ці інтеракції розподілені на приблизно 4,93 мільйона сесій, фіксують

різноплановий спектр поведінкових патернів споживачів, зокрема й перегляди сторінок, пошукові запити на продукти, операції додачі в кошик і фінальні купівельні операції. Окрім того, набір даних містить детальні продуктові метадані, які охоплюють категоріальні ієрархії продуктів, цінову політику, сегментовану на 10 квантилів і векторизовані репрезентації як текстових описів продуктів, так і їх зображень.

У табл. 4 наведено чисельний огляд параметрів досліджуваного набору даних.

Розмір вузлів прямо пропорційний числу взаємодій із продуктом разом із купівлями.

Таблиця 4

Дескриптивна статистика набору даних SIGIR 2021 E-Commerce Dataset

Загальне число операцій	36 079 307
Інтерації із продуктами	10 431 611
Операції «Додати в кошик»	329 557
Купівлі	77 848
Процентилі довжини сеансу (25/50/75)	2 / 3/8

Колір вузла відокремлює продукти в категорії, підкреслює специфічні атрибути, як-от цінові рівні. Водночас позиції вузлів зумовлені силовою симуляцією, яка забезпечує кластеризацію продуктів зі схожими характеристиками або контекстно пов'язаних.

Ребра між вузлами репрезентують силу або частоту взаємодій, які виражаються, наприклад, у сумісних покупках або функціональних залежностях. Жирність ребра корелює із силою взаємних залежностей між продуктами.

Зважаючи на масштаб набору даних, автоматична кластеризація необхідна для забезпечення зрозумілості візуалізації. Дана мета досягається завдяки імплементації алгоритму розпізнавання спільності, який базується на трансформації даних про продукти в щільні, високорозмірні вектори з використанням моделі обробки натуральної мови. Цей процес позначається терміном «вкладання» та кодує кожні внутрішні семантичні відношення продуктів, фіксує детальні схожості та залежності. Кожне вкладання для даного продукту x_i репрезентується як:

$$e_i = M(x_i), \quad (5)$$

де e_i позначає високорозмірний вектор, який фіксує семантичні атрибути продукту x_i . Ці вкладання зберігають комплексну семантичну інформацію, сприяють диференціації продуктів, залежно від їхніх контекстних відношень.

Далі високорозмірні вкладання підпорядковуються фазі розмірної редукції за посередництва операції t-розподіленого стохастичного сусіднього вкладення (t-distributed Stochastic Neighbor Embedding (далі – t-SNE)) і апроксимації та проєкції рівномірного розмаїття (Uniform Manifold Approximation and Projection (далі – UMAP)):

$$e_i' = f(e_i), \quad (6)$$

де f репрезентує нелінійну функцію відображення, яка зменшує розмірність до $\mathbb{R}_k$.

Процес кластеризації надалі оптимізується з використанням моделі Діріхле, фундаментальним принципом якої є динамічне пристосовування тематичного розподілення $p(d|d)$, яке моделюється у відношенні до документа d :

$$p(d|d) = \text{Dirichlet}(\alpha_d), \quad (7)$$

у даному контексті α_d не являє собою фіксований гіперпараметр, натомість варіює залежно від кожного документа, дозволяє моделі пристосовувати свій фокус у відповідь на специфічні атрибути.

У межах кожної визначеної теми z розподілення слів точно налаштовується з метою впровадження семантичних вкладень, створює розподілення ймовірностей softmax:

$$p(w|z;\theta) = \text{softmax}(w), \quad (8)$$

де θ позначає набір параметрів, який керує розподіленням, функція softmax обчислює ймовірність належності слова w до теми z , базуючись як на семантичному контексті, так і на контекстній відповідності в межах теми. Регуляризація сприяє контролю оверфітінгу:

$$L(\theta, \phi) = -\sum_{d=1}^D \log p(d) - \sum_{k=1}^K \log p(z_k | k) + R(\theta, \phi), \quad (9)$$

де $R(\theta, \phi) = \|\theta\|^2 + \|\phi\|^2$. Дана цільова функція мінімізує негативну логарифмічну ймовірність. Отже, регуляризація накладає штрафи на великі значення параметрів, сприяє зосередженню кожної теми на менших, більш репрезентативних наборах слів.

ВИСНОВКИ

Інтерактивна візуалізація цільових даних у вебзастосунках у формі силових орієнтовних графів забезпечує динамічну репрезентацію та спрощене дослідження даних у комплексних наборах даних. Така візуалізація надає можливість урахувати відношення різних ланок набору даних і характер їх

взаємодії. Інтеграція D3.js спрощує безшовне з'єднання даних і динамічне оновлення, згідно із принципом «вхід – оновлення – вихід», який гарантує, що кожен елемент даних відповідає тиме унікальній візуальній репрезентації.

Кластеризаційні методи на базі моделей обробки натуральної мови, поєднані із просторовою редукцією за допомогою t-SNE та UMAP, сприяють додатковому вдосконаленню шляхом забезпечення групування семантично та контекстно поєднаних елементів, що має особливо велике значення у сфері електронної комерції.

ЛІТЕРАТУРА

[1] Barker T., Westfall J. Pro Data Visualization Using R and JavaScript, Analyze and Visualize Key Data on the Web. Berkeley : Apress, CA, 2022. 274 p. DOI: 10.1007/978-1-4842-7202-2.

[2] Cheong S.H., Si Y.W. Snapshot visualization of complex graphs with force-directed algorithms. *arXiv : website*. 2022. DOI: 10.48550/arXiv.2203.16867.

[3] Cheong S.H., Si Y.W., Wong R. Online force-directed algorithms for visualization of dynamic graphs. *Information Sciences*. 2021. № 556. DOI: 10.1016/j.ins.2020.12.069.

[4] Embarak O. Data Visualization: Analyze Data to Create Visualizations for BI Systems. *Data Analysis and Visualization Using Python*. 2018. P. 293–342. DOI: 10.1007/978-1-4842-4109-7_7.

[5] Gunathilaka M., Perera N., Weerasingha P. Application of neural network for automatic symbol recognition in production of electronic navigation charts from paper charts. *Journal of Navigation*. 2024. P. 1–8. DOI: 10.1017/S0373463324000201.

[6] Iglesias M. Pro D3.js: Use D.js to create maintainable, modular, and testable charts. 2019. DOI: 10.1007/978-1-4842-5203-1.

[7] Kadijevich D.M. Web Application Development Achievement: Clarifying the Relationship Between Visual GUI Design and Textual Programming. *Towards a Collaborative Society Through Creative Learning. WCCE 2022. IFIP Advances in Information and Communication Technology / T. Keane et al. (eds)*. 2023. Vol 685. Springer, Cham. Vol. 685. https://doi.org/10.1007/978-3-031-43393-1_50.

[8] Kimunge J. Project 33: Architecting highly dynamic visualizations using React, Redux, D3.js, and RxJs. Part 1. Video. 2024. DOI: 10.13140/RG.2.2.17724.86407.

[9] Kumar G. Animated web application. *International Journal for Research in Applied Science and Engineering Technology*. 2024. № 12. P. 442–446. DOI: 10.22214/ijraset.2024.64992.

[10] Kuo L., Chang T., Lai C.C. Application of visual colors in dynamic web page design through affective cognition. *Multimedia Tools and Applications*. 2022. № 81. DOI: 10.1007/s11042-021-11732-z.

[11] Li Q., Ni L., Bai L. Display space and distribution visualization of intelligent algorithms and 3D interactive imaging technology. *Computing and*

Informatics. 2024. № 43. P. 993–1011. DOI: 10.31577/cai_2024_4_993.

[12] Lu J., Si Y.W. Clustering-based force-directed algorithms for 3D graph visualization. *The Journal of Supercomputing*. 2020. № 76. DOI: 10.1007/s11227-020-03226-w.

[13] Mathaisel D. Using data visualizations as information communication tools during a crisis: a critical review. *Journal of Information, Communication and Ethics in Society*. 2024. № 22. DOI: 10.1108/JICES-06-2023-0080.

[14] Moreb M. Design and implementation of software engineering for modern web applications. 2024. DOI: 10.4018/978-1-6684-4844-1.

[15] Morth E., Bruckner S., Smit N. ScrollyVis: Interactive Visual Authoring of Guided Dynamic Narratives for Scientific Scrollytelling. *IEEE Transactions on Visualization and Computer Graphics*. 2022. P. 1–12. DOI: 10.1109/TVCG.2022.3205769.

[16] Nehls S., Lüde J. Interactive Datenvisualisierung im Web mit D3.js. 2020. DOI: 10.1007/978-3-658-29562-2_6.

[17] Suh A., Hajij M., Wang B., Scheidegger C., Rosen P. Persistent homology guided force-directed graph layouts. *IEEE Transactions on Visualization and Computer Graphics*. 2019. P. 1–1. DOI: 10.1109/TVCG.2019.2934802.

[18] Wang S., Wang S., Fan Y., Li X., Liu Y. Leveraging large vision language model for better automatic web GUI testing. 2024. DOI: 10.48550/arXiv.2410.12157.

[19] Zhang Q. The impact of interactive data visualization on decision-making in business intelligence. *Advances in Economics, Management and Political Sciences*. 2024. № 87. P. 166–171. DOI: 10.54254/2754-1169/87/20241056.

[20] Zhong F., Xue M., Zhang J., Zhang F., Ban R., Deussen O., Wang Y. Force-directed graph layouts revisited: A new force based on the T-distribution. 2023. DOI: 10.48550/arXiv.2303.03964.

REFERENCES

[1] Barker, T., & Westfall, J. (2022). *Pro data visualization using R and JavaScript: Analyze and visualize key data on the web*. Berkeley, CA: Apress. <https://doi.org/10.1007/978-1-4842-7202-2> [in English].

[2] Cheong, S.H., & Si, Y.W. (2022). Snapshot visualization of complex graphs with force-directed algorithms. *arXiv*. <https://doi.org/10.48550/arXiv.2203.16867> [in English].

[3] Cheong, S.H., Si, Y.W., & Wong, R. (2021). Online force-directed algorithms for visualization of dynamic graphs. *Information Sciences*, 556. <https://doi.org/10.1016/j.ins.2020.12.069> [in English].

[4] Embarak, O. (2018). Data visualization: Analyze data to create visualizations for BI systems. *Data analysis and visualization using Python* (pp. 293–342). https://doi.org/10.1007/978-1-4842-4109-7_7 [in English].

[5] Gunathilaka, M., Perera, N., & Weerasingha, P. (2024). Application of neural network for automatic symbol recognition in production of electronic navigation charts from paper charts. *Journal of Navigation*,

77, 1–8. <https://doi.org/10.1017/S0373463324000201> [in English].

[6] Iglesias, M. (2019). *Pro D3.js: Use D3.js to create maintainable, modular, and testable charts*. Apress. <https://doi.org/10.1007/978-1-4842-5203-1> [in English].

[7] Kadijevich, D.M. (2023). Web application development achievement: Clarifying the relationship between visual GUI design and textual programming. In T. Keane, C. Lewin, T. Brinda, & R. Bottino (Eds.), *Towards a collaborative society through creative learning. WCCE 2022. IFIP advances in information and communication technology* (Vol. 685). Springer. https://doi.org/10.1007/978-3-031-43393-1_50 [in English].

[8] Kimunge, J. (2024). Project 33: Architecting highly dynamic visualizations using React, Redux, D3.js, and RxJs: Part 1 [Video]. <https://doi.org/10.13140/RG.2.2.17724.86407> [in English].

[9] Kumar, G. (2024). Animated web application. *International Journal for Research in Applied Science and Engineering Technology*, 12, 442–446. <https://doi.org/10.22214/ijraset.2024.64992> [in English].

[10] Kuo, L., Chang, T., & Lai, C.C. (2022). Application of visual colors in dynamic web page design through affective cognition. *Multimedia Tools and Applications*, 81. <https://doi.org/10.1007/s11042-021-11732-z> [in English].

[11] Li, Q., Ni, L., & Bai, L. (2024). Display space and distribution visualization of intelligent algorithms and 3D interactive imaging technology. *Computing and Informatics*, 43, 993–1011. https://doi.org/10.31577/cai_2024_4_993 [in English].

[12] Lu, J., & Si, Y.W. (2020). Clustering-based force-directed algorithms for 3D graph visualization. *The Journal of Supercomputing*, 76. <https://doi.org/10.1007/s11227-020-03226-w> [in English].

[13] Mathaisel, D. (2024). Using data visualizations as information communication tools during a crisis: A critical review. *Journal of Information, Communication and Ethics in Society*, 22. <https://doi.org/10.1108/JICES-06-2023-0080> [in English].

[14] Moreb, M. (2024). *Design and implementation of software engineering for modern web applications*. IGI Global. <https://doi.org/10.4018/978-1-6684-4844-1> [In English].

[15] Morth, E., Bruckner, S., & Smit, N. (2022). ScrollyVis: Interactive visual authoring of guided dynamic narratives for scientific scrollytelling. *IEEE Transactions on Visualization and Computer Graphics*, 1–12. <https://doi.org/10.1109/TVCG.2022.3205769> [in English].

[16] Nehls, S., & Lüde, J. (2020). Interaktive Datenvisualisierung im Web mit D3.js. In *Technologien für interaktive Anwendungen*. Springer. https://doi.org/10.1007/978-3-658-29562-2_6 [In English].

[17] Suh, A., Hajij, M., Wang, B., Scheidegger, C., & Rosen, P. (2019). Persistent homology guided force-directed graph layouts. *IEEE Transactions on Visualization and Computer Graphics*, 1–1. <https://doi.org/10.1109/TVCG.2019.2934802> [in English].

[18] Wang, S., Wang, S., Fan, Y., Li, X., & Liu, Y. (2024). Leveraging large vision language model for better automatic web GUI testing. *arXiv*. <https://doi.org/10.48550/arXiv.2410.12157> [in English].

[19] Zhang, Q. (2024). The impact of interactive data visualization on decision-making in business intelligence. *Advances in Economics, Management and Political Sciences*, 87, 166–171. <https://doi.org/10.54254/2754-1169/87/20241056> [in English].

[20] Zhong, F., Xue, M., Zhang, J., Zhang, F., Ban, R., Deussen, O., & Wang, Y. (2023). Force-directed graph layouts revisited: A new force based on the T-distribution. *arXiv*. <https://doi.org/10.48550/arXiv.2303.03964> [In English].

ABSTRACT

Zozulia Y. Creating interactive visualizations in web-applications

Purpose. The purpose of the article is to develop methods for interactive data visualization in web application interfaces based on force-oriented graphs. The article presents the process of developing and implementing interactive data visualization in web applications, focusing on force-oriented graphs as a means of dynamic representation and analysis of complex data sets.

Methodology. The main approach was the implementation of the "input – update – output" pattern, which provides dynamic binding of data to DOM elements to create, update and delete visual elements according to changes in the data set. As an applied method, force-oriented graphs are implemented, where the behavior of nodes is modeled according to physical principles. The forces of attraction and repulsion, described by Hooke's and Coulomb's laws, are taken into account to maintain optimal distances between nodes, and positional forces are applied to keep graphs within the viewing area.

Results. It was determined that force-oriented graphs consisting of multiple disjoint subgraphs are modeled using physical principles that govern the behavior of their constituent elements. These visualizations balance repulsive and attractive forces to maintain spatial clarity and avoid node overlap, while

applying positional adjustments to ensure that all subgraphs remain in a single view. The paper demonstrates the need for integration with the D3.js library, which facilitates data binding to the Document Object Model (DOM). This integration is based on the "input – update – output" pattern, which allows for the dynamic creation, modification, and deletion of visual elements to represent new data sets. The pattern ensures that each data element is uniquely represented and is constantly updated in response to changes, achieving robust agreement between data and visualization.

The scientific novelty lies in the fact that the order of application of advanced clustering methods for improved data organization was determined. High-dimensional embeddings derived from natural language processing models were used to capture semantic and contextual relationships in the datasets. These embeddings were further optimized using dimensionality reduction methods such as t-SNE and UMAP, which preserve meaningful relationships while simplifying the data structure. It was shown that the resulting clusters need to be represented visually to highlight semantic connectivity, improve interpretation, and aid in the discovery of hidden patterns.

Practical relevance. To demonstrate the practical utility of the proposed methodology, it was applied to the e-commerce dataset from the SIGIR 2021 e-commerce workshop. This dataset encompasses millions of interactions and extensive metadata, providing real-world context for evaluating the effectiveness of visualization. This approach enables the clustering of products based on shared characteristics, facilitating the understanding of consumer behavior and product relationships.

Keywords: interactive representation, force-oriented graphs, data linking, d3.js integration, clustering methods, dimensionality reduction.

AUTHOR'S NOTE:

Zozulia Yuliia, Postgraduate Student at the Department of Design and Technology, Kyiv National University of Culture and Arts, Kyiv, Ukraine, e-mail: julja.sotnikova@gmail.com, orcid: 0009-0008-4075-0267.

Стаття подана до редакції 02.01.2025 р.