

ЗАСТОСУВАННЯ БІБЛІОТЕК PYTHON ДЛЯ ПРОГНОЗУВАННЯ ВЛАСТИВОСТЕЙ ЕНЕРГОЕФЕКТИВНИХ ПОЛІМЕРНИХ КОМПОЗИЦІЙ

Новак Д.С. – к.т.н., доц., novak.ds@knutd.edu.ua

Київський національний університет технологій та дизайну

Метою роботи є дослідження та систематизація можливостей застосування бібліотек Python для прогнозування фізико-хімічних властивостей енергоефективних полімерних композицій, а також оцінка ефективності різних алгоритмів машинного навчання для скорочення часу розробки нових полімерних композиційних матеріалів з заданими характеристиками.

Сучасна матеріалознавча інженерія активно використовує методи машинного навчання для прогнозування властивостей нових матеріалів. Особливої уваги набуває розробка енергоефективних полімерних композицій, які знаходять застосування в сонячних панелях, теплоізоляційних матеріалах та електронних пристроях. Мова програмування Python з її потужними бібліотеками надає дослідникам універсальний інструментарій для вирішення цих задач (рис. 1).

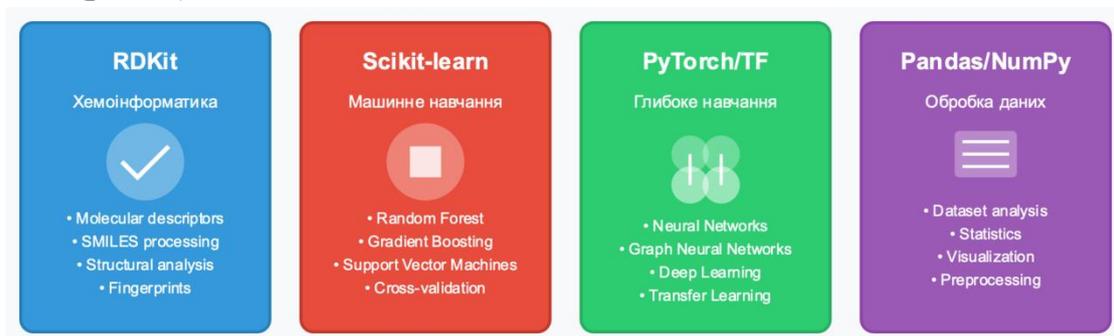


Рисунок 1 – Основні бібліотеки Python для аналізу полімерів

Екосистема Python включає спеціалізовані бібліотеки для роботи з хімічними даними та побудови прогнозних моделей (рис. 2).



Рисунок 2 – Процес прогнозування властивостей полімерів

RDKit – бібліотека для хемоінформатики, що дозволяє обробляти молекулярні структури, обчислювати дескриптори та аналізувати хімічні властивості полімерів. Вона забезпечує конвертацію SMILES-нотації в молекулярні графи та розрахунок понад 200 молекулярних дескрипторів.

Scikit-learn – головний інструмент для побудови моделей машинного навчання. Містить алгоритми регресії (Random Forest, Gradient Boosting, SVR), які використовуються для прогнозування теплопровідності, температури склування та механічних властивостей полімерів.

Типовий робочий процес включає кілька етапів. Спочатку збирається датасет з експериментальних даних про структуру та властивості полімерів. За допомогою RDKit здійснюється перетворення молекулярних структур у числові дескриптори – це можуть бути молекулярна маса, логарифм коефіцієнта розподілу (LogP), топологічний полярний поверхневий ареал (TPSA), кількість донорів та акцепторів водневих зв'язків тощо. Далі дані нормалізуються та розділяються на навчальну і тестову вибірки. Застосовуються алгоритми машинного навчання – найчастіше ансамблеві методи (Random Forest, XGBoost), які демонструють високу точність для невеликих датасетів. Для складних залежностей використовуються глибокі нейронні мережі.

Головні переваги Python-підходу: швидкість розробки прототипів, можливість інтеграції з квантово-хімічними розрахунками, відкритий доступ до бібліотек та потужна спільнота розробників. Час скринінгу скорочується з кількох місяців експериментів до годин обчислень. Основні обмеження пов'язані з якістю та обсягом навчальних даних. Для полімерних композиційних матеріалів характерна складна залежність властивостей від молекулярної архітектури, що вимагає врахування стереохімії, кристалічності та морфології. Перспективним напрямком є розробка графових нейронних мереж (Graph Neural Networks), які природно представляють молекулярну структуру. Активний розвиток набувають методи активного навчання (Active Learning), де модель сама пропонує найбільш інформативні експерименти, та трансферне навчання (Transfer Learning) для перенесення знань між спорідненими класами полімерів.

Висновки. Бібліотеки Python формують потужну екосистему для «in silico» дизайну енергоефективних полімерів. Комбінація хемоінформатичних інструментів (RDKit) з алгоритмами машинного навчання (Scikit-learn, PyTorch) дозволяє значно прискорити розробку нових композиційних матеріалів. Інтеграція експериментальних даних з обчислювальними методами відкриває шлях до створення полімерних композицій з заданими властивостями, що є критично важливим для енергетики майбутнього. Ключовими перевагами підходу є скорочення часу розробки, можливість скринінгу тисяч зразків без синтезу, висока точність прогнозів та значна економія матеріальних ресурсів. Python-екосистема забезпечує доступність цих технологій для широкого кола дослідників завдяки відкритому коду та активній спільноті розробників.