

ПОБУДОВА РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ З ВИКОРИСТАННЯМ МЕТА-НАВЧАННЯ НА ОСНОВІ OPENML

Д. Є. Осіпчук¹

¹ Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»,
Фізико-технічний інститут

Анотація

В даній роботі пропонується механізм автоматичної рекомендації алгоритмів машинного навчання та їх параметрів. Для побудови мета-атрибутів використовується база даних OpenML. В якості рекомендацій використовується реалізація механізму логічних висновків HeaRTDroid. Набір створених рекомендацій покращується за допомогою інструменту Auto-WEKA.

Ключові слова: мета-навчання, рекомендаційна система, вибір алгоритму, OpenML, Auto-WEKA, Amelia-II

Вступ

Мета-навчання (також відоме як «метанавчання» або «навчання навчатись») – стара неформальна концепція когнітивної психології, яка зовсім нещодавно стала формальною концепцією машинного навчання. На практиці, мета-навчання може використовуватись для автоматизації людських рішень (наприклад, налаштування параметрів), а потім автоматично переглядати та оптимізувати отримані рішення на основі нового досвіду, отриманого в ході навчання [1]. Інше застосування – передача знань між пов'язаними задачами.

Ціль роботи – створити рекомендаційну систему на основі мета-навчання, яка зможе полегшити роботу користувача в сфері машинного навчання. Для створення мета-знання використовуються дані експериментів з OpenML, які пізніше перетворюються в правила. Отримані знання пізніше використовуються для співставлення мета-атрибутів нового набору даних з поточним мета-знанням для отримання, можливо, кращого набору алгоритмів. На кінцевому етапі для оптимізації параметрів даного набору алгоритмів використовується інструмент Auto-WEKA.

Даний підхід наслідує загальну архітектуру мета-навчання, яка раніше була запропонована Pavel Brazdil [2]. Ми використовуємо дані про машинне навчання із публічної онлайн платформи OpenML. При створенні мета-знання використовується алгоритм Amelia-II для заповнення відсутніх даних, що не можуть бути отримані з OpenML [3]. В якості конфігуратора правил використовується механізм логічних висновків HeaRTDroid. Для додаткового налаштування створеної рекомендації виконується оптимізація гіперпараметрів інструментами Auto-WEKA [4].

Механізм рекомендації ділиться на три етапи:

- 1) Набуття знань. Мета-знання будується лише із даних OpenML

- 2) Рекомендація. Система використовує дане мета-знання і новий набір даних для побудови набору підходящих алгоритмів
- 3) Налаштування. Виконується автоматична конфігурація отриманих алгоритмів з використанням Auto-WEKA

1. Побудова мета-знання

На даному етапі основною ціллю є створення мета-знання, що описує залежність між наборами даних і оцінкою алгоритмів машинного навчання, які були виконані на цих даних (рис. 1)

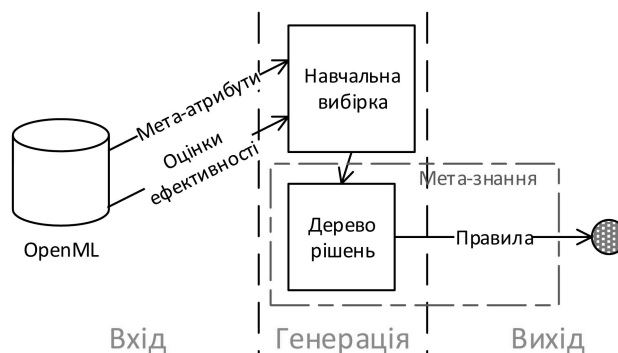


Рис. 1. Побудова мета-знання

1.1. OpenML

OpenML представляє собою відкриту інтернет-платформу, присвячену машинному навчанню. Даний сервіс дозволяє автоматизувати багато аспектів досліджень, легко інтегрується з різноманітними інструментами для машинного навчання, а також надає простий у використанні API. На сьогоднішній день нараховує близько 20 тисяч проаналізованих наборів даних.

Для кожного набору даних в базі даних OpenML доступний набір мета-атрибутів, включаючи:

- Статистична інформація (наприклад, число класів і ознак, кількість пропущених значень)
- Інформаційно-теоретичні характеристики (наприклад, ентропія класу, ознак, середнє значення ознак)
- Інформація про модель (наприклад, $J48$ або kNN_{AUC})

Кожна з даних характеристик має різний відсоток покриття ненульових значень, який коливається від 6,5% до 100%. Ми вибираємо порогове значення для покриття необхідних значень до 20% для того, щоб залишити важливі мета-атрибути. Відсутні значення заповнюються алгоритмом Amelia-II.

1.2. Amelia-II

Позначимо набір даних $(n \times k)$ як D . Алгоритм Amelia-II розбиває дані D на видимі D^{obs} та відсутні D^{mis} . Припускається, що дані відсутні випадковим чином (missing at random, MAR) [4]. З цього слідує, що шаблон відсутніх даних залежить лише від видимих даних D^{obs} . Нехай M – матриця відсутності, де $m_{ij} = 1$, якщо $d_{ij} \in D^{mis}$ і $m_{ij} = 0$ в іншому випадку. Тоді MAR визначається як

$$p(M|D) = p(M|D^{obs}) \quad (1)$$

Алгоритм припускає, що повні дані (як видимі, так і відсутні) є багатовимірними і нормальними. Нехай $\theta = (\mu, \Sigma)$ – параметри повних даних. μ – мат. очікування, Σ – матриця коваріації. Тоді ймовірність видимих даних дорівнює

$$p(D^{obs}, M|\theta) = p(M|D^{obs})p(D^{obs}|\theta) \quad (2)$$

Так як нам потрібні лише висновки стосовно параметрів повних даних, запишемо функцію подібності як

$$L(\theta|D^{obs}) \propto p(D^{obs}|\theta) \quad (3)$$

Знаючи функцію 3, знайдемо апостеріорну ймовірність як

$$p(\theta|D^{obs}) \propto p(D^{obs}|\theta) = \int p(D|\theta)dD^{mis} \quad (4)$$

Алгоритм використовує комбінацію класичного ЕМ-алгоритму та статистичний бутстреп (bootstrapping) для заповнення відсутніх даних [4].

2. Рекомендації та налаштування

Мета-знання об'єднують мета-атрибути характеристик набору даних з відповідними мітками алгоритмів. До уваги приймається лише фіксована кількість алгоритмів (у нашому випадку це перших N алгоритмів OpenML). Після цього ми фільтруємо результати за ефективністю і залишаємо лише набір найкращих алгоритмів. В результаті чого мета-знання розглядаються як помічений набір даних. Використовуючи алгоритм WEKA $J48$ створюється

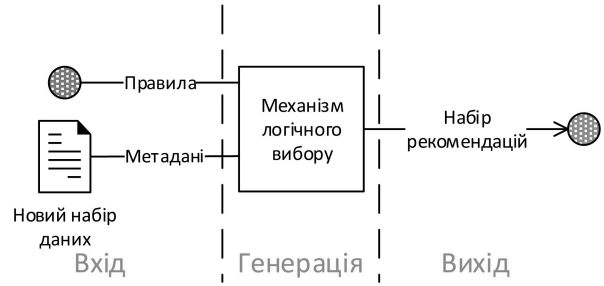


Рис. 2. Рекомендація

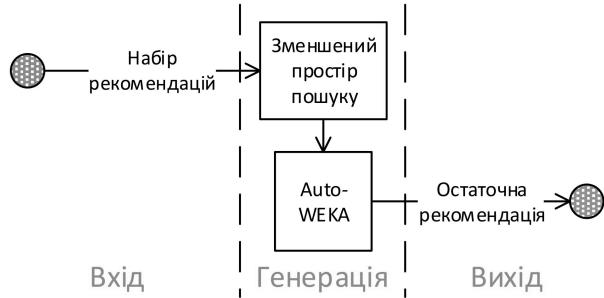


Рис. 3. Налаштування

дерево рішень, яке пізніше перетворюється в набір правил ХТТ2 (eXtended Tabular Trees). У якості конфігуратора використовується механізм логічного вибору HeaRTDroid, який представляє собою реалізацію ХТТ2. Розмір набору рекомендацій залежить від кількості правил конфігуратора. Результат рекомендації складається з імені алгоритму і набору параметрів, які відповідно конфігуратору найкраще відповідають даним (рис. 2).

На етапі налаштування ми скорочуємо простір пошуку лише до рекомендованих алгоритмів за допомогою Auto-WEKA. В результаті отримується ім'я класифікатора з одним набором параметрів (рис. 3).

Висновки

Основною метою роботи була побудова механізму вибору кращого алгоритму класифікації з використанням мета-навчання для бази даних OpenML. Даний підхід та механізм рекомендацій був протестований на 570 наборах даних. Для побудови мета-знання були використані наступні параметри:

- 1) Вибраний показник ефективності: площа під ROC-кривою ($\text{area under ROC, AUC} \geq 0.9$)
- 2) Кількість алгоритмів для розгляду: 15 найпопулярніших алгоритмів з OpenML
- 3) Використовувати мета-атрибути з менше ніж 20% відсутніх значень

В якості стандартного критерію для порівняння наших рекомендацій був вибраний метод RandomForest. Для більшості наборів даних область під ROC-кривою наших рекомендацій була більшою (401 набір даних із середнім значенням 0.044). Для 169 наборів даних AUC наших рекомендацій була нижчою (0.057)(рис. 4).

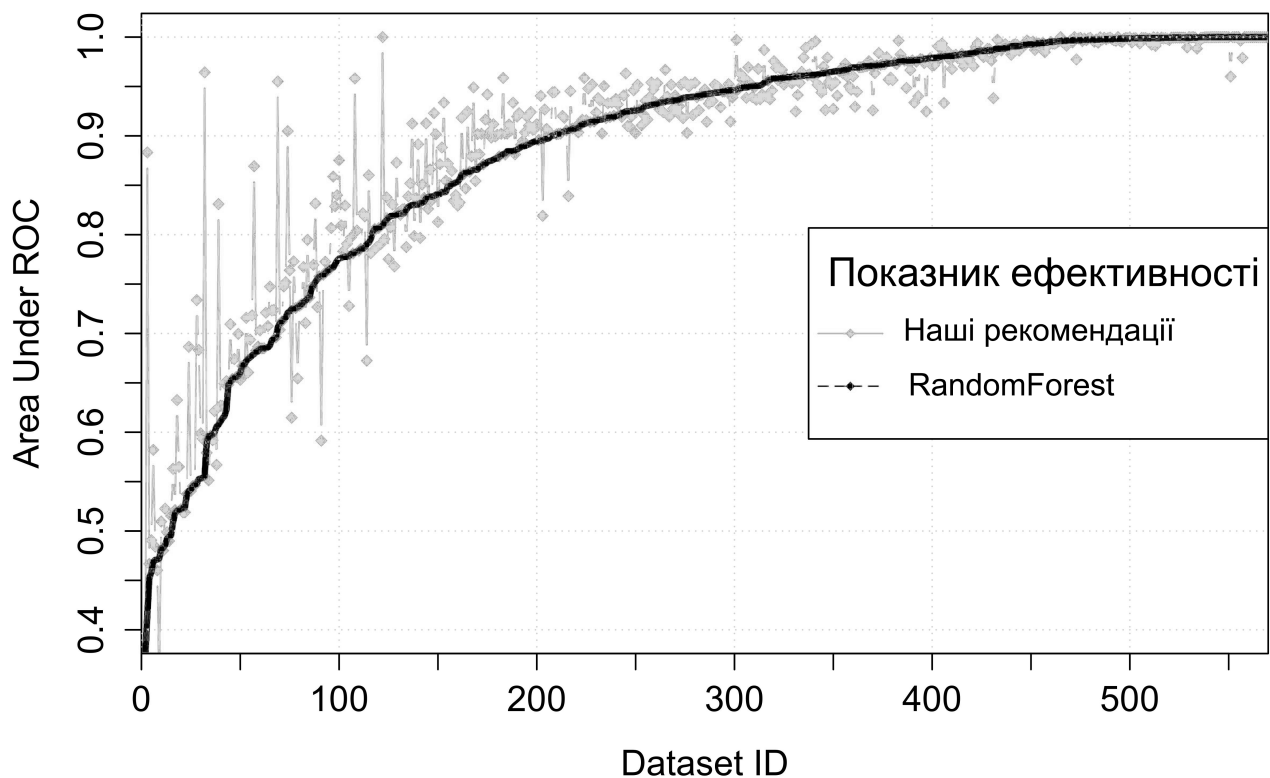


Рис. 4. Ефективність для мета-знання (AUC=90, N=15)

Перелік використаних джерел

1. Schaul T., Schmidhuber J. Metalearning // Scholarpedia. — 2010. — Vol. 5, no. 6. — P. 46–50.
2. Metalearning: Applications to Data Mining / Pavel Brazdil, Christophe Giraud-Carrier, Carlos Soares, Ricardo Vilalta. — 1 edition. — Springer Publishing Company, Incorporated, 2008. — P. 11–31. — ISBN: 3540732624, 9783540732624.
3. Honaker James, King Gary, Blackwell Matthew. Amelia II: A Program for Missing Data // Journal of Statistical Software. — 2011. — Vol. 45, no. 7. — P. 1–47.
4. Auto-WEKA: Combined Selection and Hyperparameter Optimization of Classification Algorithms / Chris Thornton, Frank Hutter, Holger H. Hoos, Kevin Leyton-Brown // Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. — KDD '13. — New York, NY, USA : ACM, 2013. — P. 847–855.