УДК: 004.65, 004.82

МЕТОД ВИБОРУ АТРИБУТІВ ЯК ЕЛЕМЕНТ МЕХАНІЗМУ ВИДОБУТКУ ЗНАНЬ

Барбарук В.М., Татарченко Г.О.

METHOD OF ATTRIBUTES SELECTION AS A ELEMENT OF KNOWLEDGE DISCOVERY IN DATABASES

Barbaruk V.M., Tatarchenko G.O.

Концепція вибору атрибутів є корисною для вирішення проблем природоохоронної науки, оскільки виявлення найбільш релевантних або прогнозованих атрибутів сприятиме зростанню економічного ефекту процесів збору та керування даними. В якості інструменту пошуку знань для оцінки структури дерева рішень, був розроблений метод вибору вузлів дерева рішень, щоб витягнути зв'язки між атрибутами та рішеннями в дереві.

У статті наводиться короткий опис концепції вибору атрибутів, деталі модернізації алгоритму вибору вузлів дерева рішень та опис програмної реалізації алгоритму для аналітичної платформи WEKA.

Ключові слова: *атрибут, дерево рішень, база знань, метод*

Вступ. Невідповідні, надмірні та невизначені особливості можуть заплутати алгоритми навчання експертних систем та змусити їх будувати погані класифікатори станів [1]. Для удосконалення результатів або чіткості були введені та випробувані дослідниками низка методів вибору атрибутів, наприклад, «Information Gain» [2] та «Relief» [3, 4]. Інші дослідники класифікаторів [5] запропонували формувати набір атрибутів, які найбільш передбачають результат без втрати первісного їх значення після зменшення кількості вагомих атрибутів.

Аналіз останніх досліджень. Методи виділення атрибутів, як правило, реалізовуються як операції пакування або фільтрації. Операції пакування формують набір атрибутів, оптимізований для даного алгоритму класифікації, котрий розглядається як чорний ящик. Багаторазово виконуючи алгоритм на різних вхідних наборах даних і вимірюючи якість набору кожного разу забезпечуються кращі результати завдяки взаємодії між пошуковим алгоритмом та схемами навчання [5]. Цей метод не є вигідним з точки зору практичного використання. Фільтрація вибирає незалежний набір атрибутів алгоритму класифікації, і ця операція займає менше часу [6]. Зазвичай, метод виділення атрибутів використовується на етапі попередньої обробки даних, щоб значно скоротити кількість атрибутів, які, наприклад, часто зустрічаються в структурованому тексті або спеціалізованих класифікаціях. Тому складність з обчислень може бути мінімізована. Але методи виділення атрибутів можуть використовуватися як інструмент пошуку знань для малих і середніх наборів даних [7], за рахунок того, що нерелевантні атрибути можуть використовуватися для підтримки операцій збору та управління даними. Технології обробки даних можуть обробляти великі набори булевих, безперервних та дискретних даних. Проте дані з природознавства, як правило, складаються з малого та середнього числа примірників та атрибутів (в кількості порядку кілька тисяч), а не десятків тисяч або мільйонів точок даних.

Мета. Деякі дослідники [8] радили використовувати прості методи, такі як широко відомий алгоритм C4.5, маючи на увазі той факт, що при зростанні кількості даних, дисперсія класичних оцінок прагне до нуля. Це означає, що малі відмінності у вибірках даних за різними атрибутами можуть виявитися статистично значимими при видобутку знань.

Модифікація алгоритму вибору вузла дерева рішень C4.5. Вивчення та застосування дерева рішень, на прикладі алгоритму C4.5 [2], практично і широко використовується як простий метод класифікації даних для індукційного аналізу [9] і описує процес прийняття рішення в зручному для розуміння вигляді.

Нехай буде згенеровано дерево рішень $T = (V, F, E, Lv, Lf)$ (рис. 1.а). Вузли представлені у вигляді $V(T) = \{v_1, ..., v_{nv}\}$, де nv - загальна кількість вузлів у дереві рішень T (за винятком листових вузлів). Нехай А являє собою набір вхідних атрибутів, де $A = \{a_1, ..., a_{na}\}$, де na - кількість атрибутів. Мітки, що

відповідають вузлам у дереві $V(T)$, представлені у вигляді $Lv = \{L(v_1), ..., L(v_{nv})\}$ і $L(v_i) \in A \ \forall \ vi \in V(T)$, де $L(v_i)$ є міткою для вузла v_i.

Не всі атрибути потрібно використовувати. Наприклад, (рис. 1 праворуч), коли існує чотири вхідних атрибути $(na = 4)$, для побудови дерева T можна використати лише два атрибути та три вузли $(nv = 3)$, тому $V(T) = \{v_1, v_2, v_3\}$, а відповідні мітки $Lv(T)$ можуть бути $\{a_1, a_2, a_2\}$, що вказує на те, що вузол v_1 помічений атрибутом a_1, а вузли v_2 та v_3 позначені одним тим же атрибутом, a_2.

Подібним чином вузли листів представлені як $F(T) = \{f_1, ..., f_{nf}\}$, де nf - кількість листових вузлів. Отже, розмір дерева рішень (T) становить $nv + nf$. Нехай C являє собою набір класів, де $C = \{c_1, ..., c_{nc}\}$, де nc - кількість класів. Мітки, що відповідають листовим вузлам $F(T)$, представлені як $Lf = \{L(f_1), ..., L(f_{nf})\}$ і $L(f_i) \in C \ \forall \ f \in F$, де $L(f_i)$ мітка для аркуша вузла f_i. Наприклад, якщо існують два вхідних класи ($nc = 2$; клас c_1 для *yes* і c_2 для *no*) і чотири листові вузли були створені як $F(T) = \{f_1, f_2, f_3, f_4\}$, відповідні мітки для $F(T)$ можуть бути позначені $L(f_i) = \{c_1, c_1, c_2, c_1\}$, що вказує на те, що f_1, f_2 і f_4 помічено з класом *yes* (c_1), а f_3 помічено з класом *no* (c_2), як це показаного на рис. 1.б. З'єднання між парами вузлів (в тому числі і для кінцевих вузлів) представлені ребрами, $E(T) = \{e_1, ..., e_{ne}\}$ де ne число ребер в T. Ребро e_i між двома вузлами $(v_j \ i \ v_k)$ визначається як $e_i = (v_j, v_k) \mid v_j, v_k \in V(T)$, а ребро e_i між вузлом v_j та вузлом листів f_k визначається як $e_i = (v_j, f_k) \mid v_j \in V(T), f_k \in F(T)$.

Позначимо через I загальну кількість правильно класифікованих екземплярів у вузлі дерева або листа, так що $I(fi)$ являє собою число правильно класифікованих екземплярів у вузлі листа f_i, а $I(v_i)$ визначається рекурсивним виразом,

$$I(v_i) = \Sigma I(f_j) \ \forall f_j \mid (v_i, f_j) \in E + \Sigma I(v_k) \ \forall v_k \mid (v_i, v_k) \in E. \quad (1)$$

Слід зауважити, що I розраховується на вузлі листа з числа класифікованих екземплярів, за мінусом кількості неправильно класифікованих екземплярів.

Наприклад,

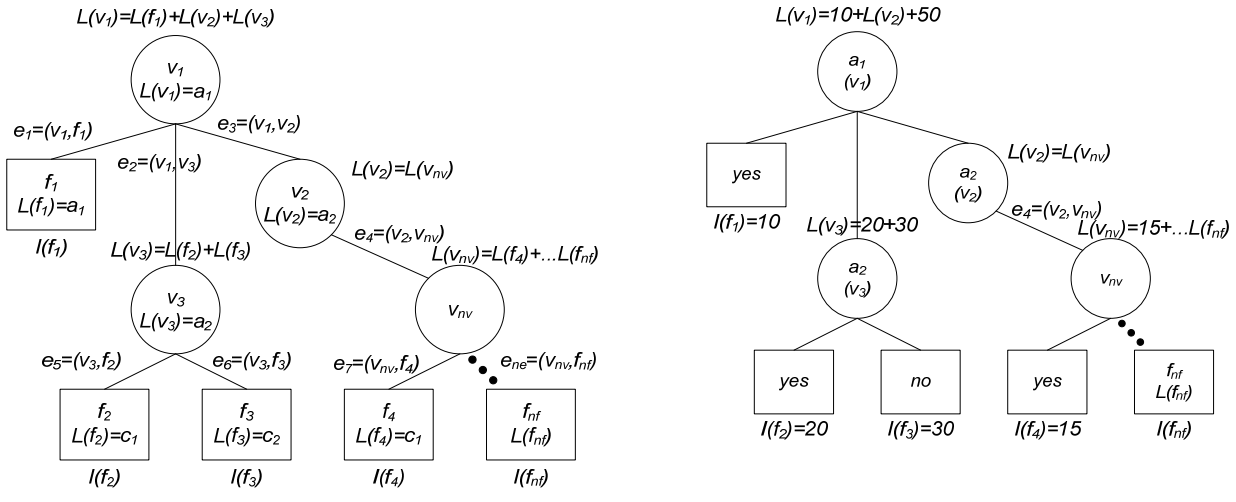
$$I(v_1) = I(f_1) \text{ де } j = 1, (v_1, f_1) \in E + I(v_2) \text{ де } k = 2, (v_1, v_2) \in E + I(v_3) \text{ де } k = 3, (v_1, v_3) \in E. \quad (2)$$

Кількість екземплярів класифікованих по шляхах, включаючи вузол v_3 обчислюється як $I(v_3) = 20 + 30$, так як $I(f_2) = 20$ і $I(f_3) = 30$. Тоді, $I(f_1) = 10$, $I(v_2) = I(v_{nv})$ і $I(v3) = 50$, так що загальна кількість примірників, класифікованих по шляхах, включаючи вузол v_1 обчислюється як $I(v_1) = 10 + I(v_{nv}) + 50$.

Таким чином, кожен атрибут a_i оцінюється за загальними екземплярами, розрахованими із загальної кількості екземплярів d вузлах, позначених a_i,

$$I(a_i) = \Sigma I(v_k) \ \forall v \ k \mid L(v_k) = a_i. \quad (3)$$

Атрибут найвищого рангу a_i має найбільше значення $I(a_i)$, що вказує на те, що, по-перше, атрибут частіше використовується в дереві рішення T, а, по-друге, що більша кількість примірників класифікується за правилами, що включають a_i для визначення класу. Слід звернути увагу на те, що рейтинг атрибутів базується на правильно класифікованих екземплярах, але і атрибути також можна класифікувати за допомогою загальних або неправильно класифікованих екземплярів.



а б

Рис. 1. а – Опис процеси вибору вузла дерева,
б – приклад дерева рішень

Реалізація алгоритму вузла дерева рішень C4.5. Алгоритм вибору вузла дерева рішень C4.5 [10] було реалізовано у вільному програмному забезпечені для аналізу даних WEKA 3.8.1, як скрипкова модифікація існуючого алгоритму J48 (рис.2).

```
public class C45 extends WekaLearner {
  private static final long serialVersionUID = 1L;
  public C45() {
    wekaCl = new J48();
    learnerName = "C4.5";
      J48 tree = new J48();
            try {
       if (!m_reducedErrorPruning)
         result = new C45PruneableClassifierTree(null,
!m_unpruned, m_CF, m_subtreeRaising,
!m_noCleanup, m_collapseTree).getCapabilities();
       else
         result = new PruneableClassifierTree(null,
!m_unpruned, m_numFolds, !m_noCleanup,
m_Seed).getCapabilities();
     }
     catch (Exception e) {
       result = new Capabilities(this);
       result.disableAll();
     }
     result.setOwner(this);      tree.setOptions(options);
      tree.buildClassifier(data);  }
  public void getParametersFromOptionsLine(String
options) {
  }
}
```

Рис. 2. Фрагмент Java-скрипта

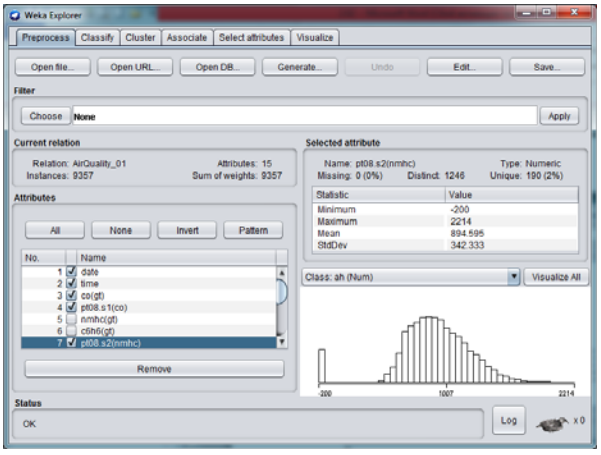


Рис. 3. Завантаження даних

Після завантаження файлу з початковими даними (рис.3) обираємо атрибути та переходимо до формування дерева рішень (рис.4) за допомогою модифікованого скрипта. Спираючись на дерево рішень (рис.5) вдається витягнути з різних, в тому числі і дуже великих, баз даних раніше невідому і достовірну інформацію, яка служить основою для прийняття рішень. Тому метод вибору вузла дерева рішень визначають також як метод підтримки прийняття рішень, заснований на пошуку та аналізі залежностей між даними.

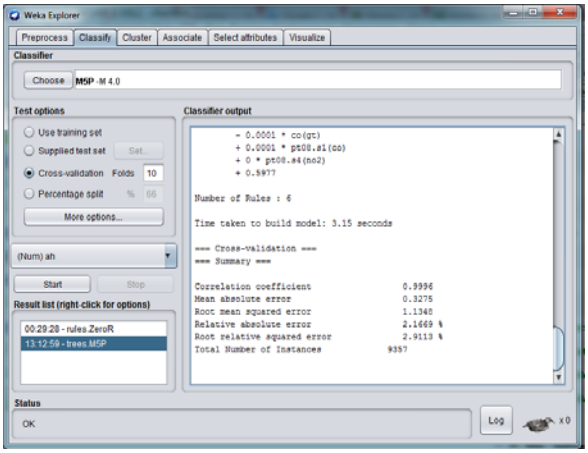


Рис. 4. Етапи генерації дерева

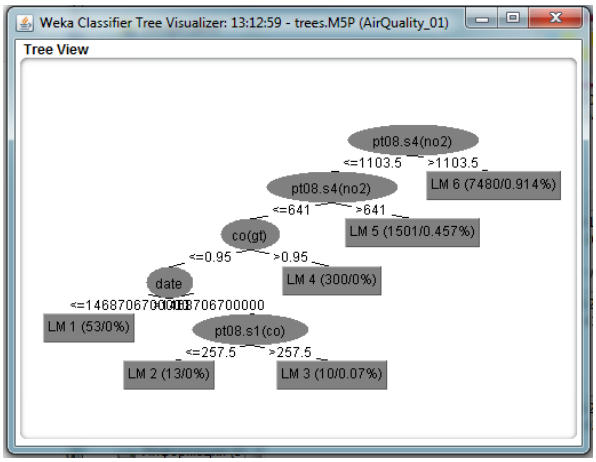


Рис. 5. Результуюче дерево рішень

Висновки. У цьому дослідженні запропоновано новий метод вибору вузлів дерева рішень, який досліджує попередньо сформоване дерево рішень як джерело інформації для вибору декількох найбільш інтелектуальних атрибутів, які допомагають будувати інші моделі. Цікавою особливістю методу вибору вузлів дерева рішень є оцінювання інформації як з попередньо згенерованих обрізаних, так і необрізаних дерев рішень. У майбутньому було б цікаво використовувати різні схеми навчання індукції дерева та вдосконалені алгоритми побудови дерева рішень, наприклад, алгоритм попереднього перегляду ID3 [6], який розраховує рентабельність розколу в вузлі, оцінюючи його вплив на більш глибокі нащадки вузла.

Наразі програмне забезпечення вибору вузлів дерева рішень - це скрипт, який запускає WEKA для генерації дерева рішень та перевірки атрибутів наборів даних.

Література

1. Last M, Kandel A, Maimon, O (2001) Information-theoretic algorithm for feature selection. Pattern Recognit lett 22: 799-811.
2. Quinlan JR (1993) C4.5: Programs for Machine Learning, Morgan Kaufmann, San Mateo.
3. Kira K, Rendell L (1992) A practical approach to feature selection. In Proc. 9 th ICML 1992, 249-256.
4. Kononenko I (1994) Estimating attributes: Analysis and Extensions of Relief, In Proc. of 7 th ECML, 171-182.
5. Hall MA, Holmes G (2003) Benchmarking Attribute Selection Techniques for Discrete Class Data Mining, IEEE Trans Knowl Eng 15: 1437-1447. (4)
6. Esmeir S, Markovitch S (2004) Lookahead-based algorithms for anytime induction of decision trees. In Proc. of 21 st ICML 2004, 257-264.
7. Jensen R, Shen Q (2007) Fuzzy-rough sets assisted attribute selection. IEEE Trans Fuzzy Syst 15: 73-89.
8. Spate JM, Gibert K, Sànchez-Marrè M, Frank E, Comas J, Athanasiadis I, Letcher R (2006) Data mining as a tool for environmental scientists. In: Voinov, A, Jakeman AJ, Rizzoli AE (eds). In Proc. of the 3 rd iEMSs, 0-22, Burlington
9. Mitchell TM (1997) Machine learning, McGraw-Hill, New York.
10. WEKA (2008) Documentation in Weka 3: Data Mining Software in Java, available at: http://www.cs.waikato.ac.nz/ml/weka/. Accessed on 28 July 2008.

References

1. Last M, Kandel A, Maimon, O (2001) Information-theoretic algorithm for feature selection. Pattern Recognit lett 22: 799-811.
2. Quinlan JR (1993) C4.5: Programs for Machine Learning, Morgan Kaufmann, San Mateo.
3. Kira K, Rendell L (1992) A practical approach to feature selection. In Proc. 9 th ICML 1992, 249-256.
4. Kononenko I (1994) Estimating attributes: Analysis and Extensions of Relief, In Proc. of 7 th ECML, 171-182.
5. Hall MA, Holmes G (2003) Benchmarking Attribute Selection Techniques for Discrete Class Data Mining, IEEE Trans Knowl Eng 15: 1437-1447. (4)
6. Esmeir S, Markovitch S (2004) Lookahead-based algorithms for anytime induction of decision trees. In Proc. of 21 st ICML 2004, 257-264.
7. Jensen R, Shen Q (2007) Fuzzy-rough sets assisted attribute selection. IEEE Trans Fuzzy Syst 15: 73-89.
8. Spate JM, Gibert K, Sànchez-Marrè M, Frank E, Comas J, Athanasiadis I, Letcher R (2006) Data mining as a tool for environmental scientists. In: Voinov, A, Jakeman AJ, Rizzoli AE (eds). In Proc. of the 3 rd iEMSs, 0-22, Burlington
9. Mitchell TM (1997) Machine learning, McGraw-Hill, New York.
10. WEKA (2008) Documentation in Weka 3: Data Mining Software in Java, available at: http://www.cs.waikato.ac.nz/ml/weka/. Accessed on 28 July 2008.

Барбарук В.Н., Татарченко Г.О. Метод выбора атрибутов как элемент механизма добычи знаний.

Концепция выбора атрибутов является полезной для решения проблем природоохранной науки, поскольку выявление наиболее релевантных или прогнозируемых атрибутов способствует росту экономического эффекта процессов сбора и управления данными. В качестве инструмента поиска знаний для оценки структуры дерева решений, был разработан метод выбора узлов дерева решений, с целью установления связей между атрибутами и решениями в дереве.

В статье представлено краткое описание концепции выбора атрибутов, детали модернизации алгоритма выбора узлов дерева решений и описана программная реализация алгоритма на базе аналитической платформы WEKA.

Ключевые слова: *атрибут, дерево решений, база знаний, метод*

Barbaruk V.M., Tatarchenko G.O. Attributes Selection Method as an Element of Knowledge Discovery in Databases

The concept of the attributes selection is useful for addressing environmental science issues, since identifying the most relevant or predicted attributes will increase the economic effect of data collection and management processes. As a search tool for evaluating the decision tree structure, a decision tree method has been developed to draw relationships between attributes and solutions in a tree.

The article gives a brief description of the concept of the attributes choice, details of the upgrade of the algorithm for choosing decision tree nodes and a description of the program implementation of the algorithm for the WEKA analytical platform.

Keywords: *attribute, decision tree, knowledge base, method*

Барбарук В.М. – к.т.н., доцент, доцент кафедри комп'ютерної інженерії Східноукраїнського національного університету імені Володимира Даля, e-mail: barbaruk.viktor@gmail.com

Татарченко Г.О. – д.т.н.. проф., завідувач кафедри міського будівництва та господарства Східноукраїнського національного університету імені Володимира Даля, email: tatarchenkogalina@gmail.com

Рецензент: д.т.н., проф. ***Рязанцев О.І.***

Стаття подана 28.08.2017