Nghiên cứu và ứng dụng cây quyết định trong bài toán tuyển dụng nhân sự

Đặng Văn Nam[1,*] , Nguyễn Thị Phương Bắc[1], Nguyễn Thị Hải Yến[1]
[1] *Trường Đại học Mỏ - Địa chất*

TÓM TẮT

Thế giới đang bước vào cuộc cách mạng 4.0, Trí tuệ nhân tạo (Artificial Intelligence), Dữ liệu lớn (Big Data), Học máy (Machine Learning), Học sâu (Deep Learning) đã được nghiên cứu, áp dụng và phục vụ cho rất nhiều lĩnh vực. Với các công ty, doanh nghiệp nói chung, đặc biệt là các công ty về công nghệ thông tin việc tuyển dụng nhân sự là việc làm mang tính chất chiến lược và được thực hiện rất thường xuyên, liên tục; Nhằm tuyển dụng được những ứng viên đáp ứng được các yêu cầu công việc. Tuy nhiên, với hàng trăm hồ sơ nộp vào cho mỗi vị trí tuyển dụng, việc sàng lọc để chọn ra được những hồ sơ ứng viên có tiềm năng không phải là một công việc dễ dàng. Trong bài báo này nhóm tác giả sẽ trình bày các nội dung về cây quyết định (Decision Tree) và việc ứng dụng cây quyết định trong một bài toán thực tế đó là hỗ trợ việc ra quyết định tuyển dụng nhân sự dựa trên cơ sở dữ liệu lịch sử tuyển dụng trước đây. Nhóm tác giả cũng tiến hành lập trình module ID3_hire_employees bằng ngôn ngữ Python sử dụng thuật toán ID3 để minh họa một cách trực quan việc xây dựng cây quyết định với một tập dữ liệu cụ thể.

Từ khóa: Cây quyết định; ID3; Học máy

1. Đặt vấn đề

Giả sử một công ty về công nghệ thông tin (CNTT) đang cần tuyển nhân viên cho vị trí lập trình viên, vị trí này được tuyển dụng liên tục nhằm đảm bảo nguồn nhân lực cho việc hoàn thành các dự án. Với mỗi một hồ sơ ứng viên tuyển dụng bao gồm 06 thuộc tính, trong đó có 05 thuộc tính quan trọng ảnh hưởng tới quyết định tuyển dụng ứng viên đó và 1 thuộc tính cho biết kết quả ứng viên đó có được tuyển dụng hay không? Chi tiết các thuộc tính của một ứng viên như trong bảng 1.

Bảng 1. Danh sách các thuộc tính của một hồ sơ

STT	Tên thuộc tính	Giá trị của thuộc tính	Ý nghĩa
1	Level	Đại học, Cao đẳng, trung tâm tin học	Cho biết trình độ đào tạo của ứng viên
2	Job	Có, không	Cho biết ứng viên đó hiện tại đang đi làm hay không?
3	N_Company	0,1,2…	Là một số cho biết số lượng công ty mà ứng viên đó đã từng làm.
4	Top_Train	Có, không	Cho biết trường mà ứng viên đó được đào tạo có nằm trong số các trường hàng đầu về CNTT hay không?
5	Project	Có, không	Cho biết ứng viên đã từng tham gia vào một dự án thực tế liên quan không?
6	Result	Có, không	Cho biết kết quả ứng viên đó có được tuyển dụng (có) hay không được tuyển dụng (không) vào công ty?

Bảng 2 bên dưới bao gồm 15 bộ dữ liệu cho biết kết quả tuyển dụng của công ty cho các lần trước đây, đây chính là dữ liệu lịch sử tuyển dụng (Training data) của công ty cho vị trí này. Câu hỏi đặt ra là với nhu cầu tuyển dụng hiện nay của công ty sẽ có rất nhiều hồ sơ ứng viên được nộp vào để đăng ký, do đó để hỗ trợ cho việc phân lớp một ứng viên dựa vào các thuộc tính đã đề cập ở trên trên cơ sở dữ liệu lịch sử tuyển dụng trước đây để hỗ trợ việc ra quyết định xem ứng viên đó sẽ được phân vào lớp được tuyển dụng hay không được tuyển dụng.

* *Tác giả liên hệ*
Email: dangvannam@humg.edu.vn

Bảng 2. Tập dữ liệu lịch sử tuyển dụng nhân sự (Training data)

STT	ID hồ sơ	Các thuộc tính chính của hồ sơ ứng viên					Result
		Level	Job	N_Company	Top_Train	Project	
1	0175	Đại học	Có	2	Không	Không	Có
2	0217	Đại học	Không	1	Có	Có	Có
3	0222	Cao đẳng	Không	4	Không	Không	Không
4	0310	Đại học	Có	1	Có	Không	Có
5	0343	Đại học	Có	5	Có	Không	Có
6	0356	Đại học	Không	1	Có	Có	Có
7	0432	Cao đẳng	Có	0	Không	Không	Không
8	0477	Đại học	Không	6	Không	Có	Có
9	0489	Trung tâm tin học	Có	2	Không	Có	Có
10	0490	Cao đẳng	Không	3	Không	Không	Không
11	0551	Đại học	Không	0	Có	Không	Có
12	0563	Trung tâm tin học	Có	3	Không	Có	Có
13	0742	Đại học	Không	0	Không	Không	Không
14	0777	Trung tâm tin học	Có	4	Không	Không	Không
15	0812	Cao đẳng	Không	5	Không	Có	Có

Giả sử có 3 hồ sơ ứng viên mới nộp vào công ty để đăng ký tuyển dụng (Bảng 3), chúng ta đã biết các thuộc tính Level, Job, N_Company, Top_Train, Project, và chưa biết được giá trị của thuộc tính Result. Yêu cầu đặt ra là dựa trên cơ sở dữ liệu Training data dự đoán giá trị cho thuộc tính Result xem hồ sơ nào có được tuyển dụng hay không được tuyển dụng vào công ty.

Bảng 3. Dữ liệu hồ sơ ứng viên mới (Testing data)

STT	ID hồ sơ	Các thuộc tính chính của hồ sơ ứng viên					Result
		Level	Job	N_Company	Top_Train	Project	
1	1001	Đại học	Có	3	Không	Không	Có/không?
2	1002	Cao đẳng	Có	2	Có	Có	Có/không?
3	1003	Trung tâm tin học	Có	2	Có	Không	Có/không?

Trong nội dung của bài báo, nhóm tác giả sẽ nghiên cứu về cây quyết định và ứng dụng giải thuật cây quyết định trong việc phân lớp hồ sơ tuyển dụng, nhằm mục đích hỗ trợ việc ra quyết định; Nhóm tác giả sẽ tiến hành tính toán và xây dựng cây quyết định sử dụng thuật toán Iterative Dichotomiser - ID3 với tập dữ liệu mẫu được cho trong bảng 2 từ đó xác định tập luật tương ứng và giá trị Result cho trong bảng 3.

2. Cây quyết định

2.1. Giới thiệu về cây quyết định

Cây quyết định là một cấu trúc phân cấp của các nút và các nhánh có các tính chất sau:

- Mỗi nút trong (internal node) biểu diễn một thuộc tính cần kiểm tra giá trị (an attribute to be tested) đối với các các tập thuộc tính.
- Nút lá (leaf node) hay còn gọi là nút trả lời biểu thị cho một lớp các trường hợp mà nhãn của nó là tên của lớp, nó biểu diễn một lớp (a classififcation)
- Nút nhánh (branch) từ một nút sẽ tương ứng với một giá trị có thể của thuộc tính gắn với nút đó.
- Nhãn (lable) của nút này là tên của thuộc tính và có một nhánh nối nút này đến các cây con ứng với mỗi kết quả có thể có phép thử. Nhãn của nhánh này là các giá trị của thuộc tính đó. Nút trên cùng gọi là nút gốc.

Quá trình xây dựng một cây quyết định cụ thể bắt đầu bằng một nút rỗng bao gồm toàn bộ các đối tượng huấn luyện và làm như sau:

Bước1: Nếu tại nút hiện thời, tất cả các đối tượng huấn luyện đều thuộc vào một lớp nào đó thì nút này chính là nút lá có tên là nhãn lớp chung của các đối tượng.

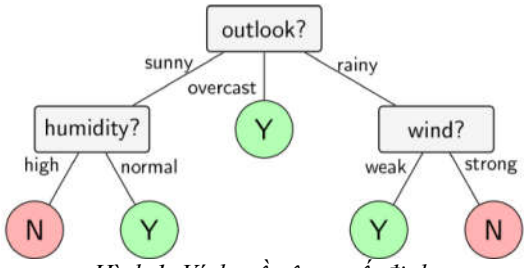
Bước 2: Trường hợp ngược lại, sử dụng một độ đo, chọn thuộc tính điều kiện phân chia tốt nhất tập mẫu huấn luyện có tại nút.

Bước 3: Tạo một nút con của nút hiện thời bằng số các giá trị khác nhau của thuộc tính được chọn.

Gán cho mỗi nhánh từ nút cha đến nút con một giá trị của thuộc tính rồi phân chia các đối tượng huấn luyện vào các nút con tương ứng

Bước 4: Nút con K được gọi là thuần nhất, trở thành lá, nếu tất cả các đối tượng mẫu tại đó đều thuộc vào cùng một lớp

Bước 5: Lặp lại bước 1 – 3 đối với mỗi nút chưa thuần nhất.



Hình 1. Ví dụ về cây quyết định

Có nhiều thuật toán để xây dựng cây quyết định như ID3, CART, J48, C4.5…Việc lựa chọn thuật toán nào để việc phân lớp đạt hiệu quả cao, kết quả đáng tin cậy phụ thuộc vào nhiều yếu tố, trong đó cấu trúc dữ liệu của các thuộc tính sẽ có ảnh hưởng lớn đến kết quả của các thuật toán.

Với dữ liệu lịch sử tuyển dụng như trong bảng 2 ta có thể chuyển đổi tất cả giá trị của các thuộc tính về cùng một dạng dữ liệu số như sau: Với thuộc tính Level: Giá trị "Đại học" ~ 2, "Cao đẳng" ~ 1; "Trung tâm tin học" ~ 0; Với các thuộc tính Job, N_Company, Top_Train, Project, Result: Giá trị "Có" ~ 1, "Không" ~ 0; Bảng 4 thể hiện kết quả chuyển đổi dữ liệu của các thuộc tính trong bảng 2 về cùng một kiểu dữ liệu dạng số.

Bảng 4. Tập dữ liệu đã chuyển đổi về dạng số

STT	ID hồ sơ	Các thuộc tính chính của hồ sơ ứng viên					Result
		Level	Job	N_Company	Top_Train	Project	
1	0175	2	1	2	0	0	1
2	0217	2	0	1	1	1	1
3	0222	1	0	4	0	0	0
...	...	…	...	...	...	...	...

Trong số các thuật toán xây dựng cây quyết định ở trên ID3 là thuật toán ra đời từ rất sớm, tương đối phổ biến và đặc biệt thuật toán này có hiệu quả phân lớp cao đối với tập dữ liệu có giá trị của thuộc tính là số. Vì vậy, nhóm tác giả lựa chọn thuật toán ID3 để tiến hành xây dựng cây quyết định cho tập dữ liệu tuyển dụng nhân sự ở trên.

2.2. Thuật toán ID3

Để xây dựng cây quyết định với thuật toán ID3 trước tiên cần xác định thứ tự của thuộc tính cần được xem xét tại mỗi bước. Trong trường hợp đối tượng có nhiều thuộc tính và mỗi thuộc tính có nhiều giá trị khác nhau thì việc xác định được thứ tự tối ưu nhất của các thuộc tính thường là rất khó. Vì vậy, để đơn giản ý tưởng của ID3 như sau:

- Thực hiện giải thuật tìm kiếm tham lam đối với không gian các cây quyết định có thể.
- Xây dựng các nút từ trên xuống (Top-Down), bắt đầu từ nút gốc.
- Ở mỗi nút, xác định thuộc tính kiểm tra là thuộc tính có khả năng phân loại tốt nhất.
- Tạo mới một cây con của nút hiện tại cho mỗi giá trị có thể của thuộc tính kiểm tra, và tập dữ liệu đầu vào sẽ được tách thành các tập con tương ứng với các cây con vừa tạo.
- Mỗi thuộc tính chỉ được phép xuất hiện tối đa 1 lần đối với bất kỳ đường đi nào trong cây.
- Quá trình phát triển cây sẽ tiếp tục cho tới khi:
  - Cây quyết định phân loại hoàn toàn các dữ liệu đầu vào.
  - Tất cả các thuộc tính tập dữ liệu được sử dụng.

Như vậy, để thuật toán ID3 thực hiện được, cần phải xác định được thứ tự chọn các thuộc tính và chọn được thuộc tính quan trọng nhất cho việc phân lớp ứng với nút đó. Tham số được sử dụng để xác định thuộc tính phân loại tốt nhất cho mỗi bước là Information Gain.

2.3. Xác định Information Gain

Information Gain được xác định thông qua Entropy của một tập.

Entropy đo mức độ hỗ tạp của một tập, Entropy của tập S đối với việc phân lớp có n lớp được xác định

như sau:

$$\text{Entropy}(S) = -\sum_{i=1}^{n} p_i * log_2(p_i)$$

Trong đó p_i là tỷ lệ các đối tượng trong tập S thuộc vào lớp i, và $0*\log_2 0 = 0$

- Entropy = 0 nếu tất các các đối tượng đều thuộc vào cùng một lớp.
- Entropy = 1 nếu các đối tượng thuộc vào các lớp có số lượng như nhau.
- Entropy ∈ (0,1) nếu các đối tượng thuộc vào các lớp khác nhau có số lượng khác nhau.

Information Gain đo mức độ giảm Entropy nếu chi tập S theo các giá trị của thuộc tính đó. Information Gain của thuộc tính A đối với tập S được tính như sau:

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{S_v}{S} Entropy(S_v)$$

Trong đó Values(A) là tập giá trị có thể của thuộc tính A, và $S_v = \{x \mid x \text{ thuộc } S, \text{ và } x_a = v\}$
Trong ID3, tại mỗi nút, thuộc tính được chọn là thuộc tính có Informaiton Gain đạt giá trị lớn nhất.

3. Kết quả xây dựng cây quyết định sử dụng thuật toán ID3

Nhóm tác giả sử dụng bộ công cụ Enthought Canopy; Các gói (package): Jupyter, pydotplus; Thư viện hỗ trợ đồ họa: GraphViz; Thư viện giải thuật: sklearn; Tệp dữ liệu lịch sử tuyển dụng được lưu trữ trong tệp Hosotd_data.csv.

Modul ID3_hire_employees được xây dựng bằng ngôn ngữ Python, Quá trình xây dựng cây quyết định đối với bài toán tuyển dụng sử dụng thuật toán ID3 được thực hiện như sau:

• Trước tiên tiến hành đọc dữ liệu lịch sử truyển dụng trong tệp Hosotd_data.csv và chuyển đổi dữ liệu của các thuộc tính về dạng số để cho thuật toán ID3 thực hiện được. Kết quả chạy được mô tả như trong hình 2 dưới đây.

• Bước 2 cần xác định đâu là thuộc tính chính tham gia vào việc xây dựng cây quyết định (Level, Job, N_Company, Top_Train, Project), và đâu là thuộc tính phân lớp (Result). (Hình 3)

In [1]:
```
import numpy as np
import pandas as pd
from sklearn import tree
from IPython.display import Image
from sklearn.externals.six import StringIO
import pydotplus

#Đọc dữ liệu đầu vào từ file excel
input_tuyendung = "E:/All_NCKH/Baibao_Hoinghi/ERSD2018/Code/Hosotd_data.csv"
datafile = pd.read_csv(input_tuyendung, header = 0)
datafile.head()

#Thực hiện chuyển đổi dữ liệu các thuộc tính về dạng số
d = {'Dai hoc': 2, 'Cao dang':1,'Trung tam tin hoc': 0}
datafile['Level'] = datafile['Level'].map(d)
d = {'Co': 1, 'Khong': 0}
datafile['Job'] = datafile['Job'].map(d)
datafile['Top_Train'] = datafile['Top_Train'].map(d)
datafile['Project'] = datafile['Project'].map(d)
datafile['Result'] = datafile['Result'].map(d)
datafile.head()
```

Out[1]:

	Level	Job	N_Company	Top_Train	Project	Result
0	2	1	2	0	0	0
1	2	0	1	1	1	1

Hình 2. Đọc file mẫu và chuyển đổi dữ liệu của các thuộc tính về dạng số

In [2]:
```
#Xác định các thuộc tính tham gia vào xây dựng cây quyết định
field = list(datafile.columns[:5])

axis_y = datafile["Result"] #Xác định thuộc tính phân lớp
axis_X = datafile[field]

field
```

Out[2]: `['Level', 'Job', 'N_Company', 'Top_Train', 'Project']`

Hình 3. Xác định thuộc tính tham gia và thuộc tính phân lớp của cây quyết định

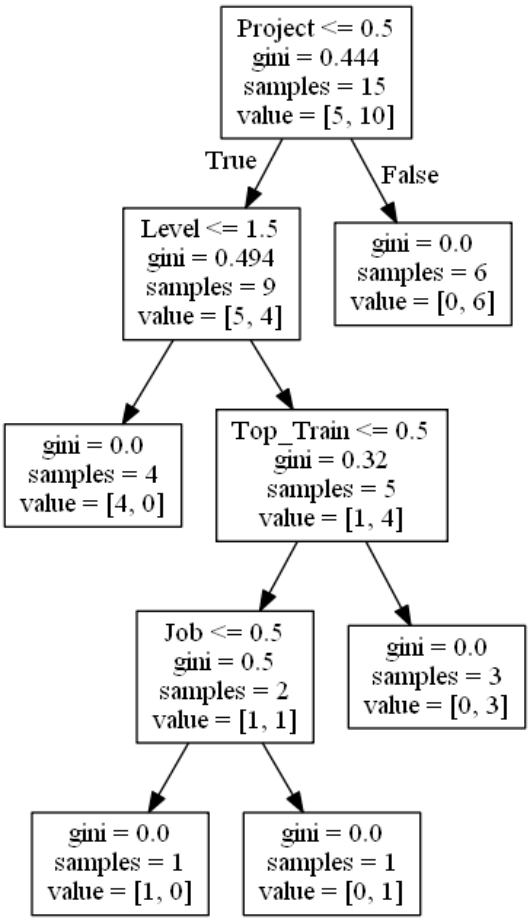
• Sử dụng thư viện sklearn để tiến hành xây dựng cây quyết định theo thuật toán ID3, và thư viện đồ

họa GraphViz để hiển thị kết quả.

```
In [3]: #Xây dựng cây quyết định
        datafile = tree.DecisionTreeClassifier()
        datafile = datafile.fit(axis_X,axis_y)

        #Hiển thị cây quyết định dạng đồ họa
        dot_data1 = StringIO()
        tree.export_graphviz(datafile, out_file=dot_data1,
                             feature_names=field)
        Tree_graph = pydotplus.graph_from_dot_data(dot_data1.getvalue())
        Image(Tree_graph.create_png())
```

Hình 4. Xây dựng cây quyết định với tập dữ liệu mẫu sử dụng giải thuật ID3



Hình 5. Cây quyết định ứng với tập dữ liệu mẫu

Với cây quyết định được xây dựng như trên, tập luật sinh ra để dự đoán quyết định có/không được tuyển dụng như sau:

1) *If(Project=True) then Result=True*
2) *If(Project=False) AND (Level<2) then Result=False*
3) *If(Project=False) AND (Level=2) AND (Top_Train=True) then Result=True*
4) *If(Project=False) AND (Level=2) AND (Top_Train=False) AND (Job=True) Then Result=True*

Dựa vào tập luật ở trên, có thể dự đoán được kết quả cho tập Testing data như dưới đây:

Bảng 4. Giá trị của thuộc tính Result cho tập Testing data

STT	ID hồ sơ	Level	Job	N_Company	Top_Train	Project	Result
1	1001	Đại học	Có	3	Không	Không	Có
2	1002	Cao đẳng	Có	0	Có	Có	Có
3	1003	Trung tâm tin học	Có	2	Có	Không	Không

4. Kết luận

Học máy đang được nghiên cứu và ứng dụng trong rất nhiều lĩnh vực, trong đó cây quyết định (Decision tree) cùng với K-means, SVN (Support Vector Machines),... là những giải thuật rất quan trọng của học máy. Không chỉ với học máy, Cây quyết định còn là một trong những mô hình dự đoán được sử dụng phổ biến trong khai phá dữ liệu (Data mining). Trong nội dung của bài báo này nhóm tác giả đã tiến hành nghiên cứu về giải thuật cây quyết định, đồng thời sử dụng cây quyết định cho một bài toán cụ thể đó là tuyển dụng nhân sự và tiến hành lập trình module id3_hire_employees xây dựng cây quyết định với thuật toán ID3 bằng ngôn ngữ Python cho tập dữ liệu mẫu với 15 bản ghi. Kết quả xây dựng cây quyết định được trình bày một cách trực quan như trong hình 5; Có thể thấy rằng cây quyết định là một phương pháp để phân lớp các đối tượng khá hiệu quả và dễ hiểu. Tuy nhiên để đảm bảo hiệu quả và độ tin cậy của các quyết định thì tập dữ liệu mẫu (Training data) phải đủ lớn và đáng tin cậy khi đó các tập luật được sinh ra mới là các tập luật tốt. Như tập dữ liệu mẫu ở trên chỉ với 15 bản ghi thì hiệu quả ứng dụng cây quyết định để dự đoán các trường hợp khác là không cao.

Tài liệu tham khảo

Thân Văn Khoát, 2017, *Bải giảng Học Máy (Machine Learning)*, Viện công nghệ thông tin và Truyền thông, ĐH Bách khoa Hà Nội.

Quinlan, J.R, 1986, *Induction of Decision trees*, Mach. Learn. 1, 81-106

Ian H.Witten, Eibe Frank, 2005, *Data Mining: Practical Machine Learning Tools and Techniques*, Margan Kaufmann

ABSTRACT

Research and application of decision tree in the recruitment problem

Dang Van Nam[1], Nguyen Thi Phuong Bac[1], Nguyen Thi Hai Yen[1]

[1] *Hanoi University of Mining and Geology*

The content of this article will discuss the application of decision trees to support decision making on recruitment based on historical recruitment history data. In addition, the team will illustrate the decision tree construction for the recruiment problem with a specific data set using the ID3 algorithm.

Keywords: Decision tree, ID3, Machine Learning.