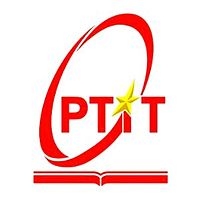
**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**Cơ Sở Tại TP. HỒ CHÍ MINH**

**🙣o0o🙡**



**PHÁT TRIỂN HỆ THỐNG DỰA TRÊN TRI THỨC**

Xây dựng kho dữ liệu và phát triển ứng dụng hỗ trợ tư vấn lương phù hợp với nghề nghiệp và trình độ

*Giảng viên hướng dẫn*: **Nguyễn Ngọc Duy**

*Lớp*: **D16CQIS01**

*Sinh viên thực hiện*: **NHÓM 1**

* *Vũ Thị Ngoãn* N16DCCN105
* *Trần Đức Vân Khanh* N16DCCN074
* *Lê Mỹ Linh* N16DCCN085
* *Nguyễn Phúc Nguyên* N16DCCN108
* *Nguyễn Đông Nhật* N16DCCN111
* *Nguyễn Trọng Tài*  N15DCCN055

TP. Hồ Chí Minh, ngày 26 tháng 10 năm 2020

***Mục lục***

[I. Giới thiệu đề tài 3](#_Toc54651434)

[1. Tên đề tài và lý do chọn đề tài 3](#_Toc54651435)

[2. Mục đích xây dựng đề tài 3](#_Toc54651436)

[II. Xây dựng kho dữ liệu 4](#_Toc54651437)

[1. Quy trình xây dựng mô hình khai phá dữ liệu: 4](#_Toc54651438)

[2. Giới thiệu cấu hình quá trình bộ gom cụm phân lớp: 5](#_Toc54651439)

[a. Giới thiệu về phân lớp: 5](#_Toc54651440)

[b. Quy trình phân lớp: 6](#_Toc54651441)

[c. Các kỹ thuật phân lớp: 7](#_Toc54651442)

[d. Cây quyết định 8](#_Toc54651443)

[e. Các công thức: 10](#_Toc54651444)

[3. Đặc tả dữ liệu: 11](#_Toc54651445)

[4. Sử dụng Weka để chọn được bộ dữ liệu ( Ổn định dữ liệu ) 14](#_Toc54651446)

[5. Bộ dữ liệu gốc: 17](#_Toc54651447)

[6. Tiến hành xây dựng và đánh giá mô hình dữ liệu 18](#_Toc54651448)

[a. Giải thuật NaiveBayes 18](#_Toc54651449)

[b. Giải thuật ZeroR 19](#_Toc54651450)

[c. Giải thuật J48 với độ tin cậy là 0.25: 20](#_Toc54651451)

[d. Giải thuật J48 với độ tin cậy là 0.5: 21](#_Toc54651452)

[7. Chọn cấu hình chạy giải thuật: 22](#_Toc54651453)

[8. Tạo bộ dữ liệu thứ 2: 25](#_Toc54651454)

[III. Test dữ liệu 27](#_Toc54651455)

[1. Test dữ liệu lần thứ nhất: 27](#_Toc54651456)

[2. Test dữ liệu lần thứ hai : 29](#_Toc54651457)

[a. Tạo bộ dữ liệu test đúng với dữ liệu gốc 29](#_Toc54651458)

[b. Test dữ liệu khuyết: 34](#_Toc54651459)

[c. Test bộ dữ liệu sai với dữ liệu gốc 35](#_Toc54651460)

[3. Trên chương trình: 41](#_Toc54651461)

[a. Chạy trên bộ dữ liệu gốc: 41](#_Toc54651462)

[b. Một số trường hợp chứng minh: 47](#_Toc54651463)

[c. Trên bộ dữ liệu thứ 2: 50](#_Toc54651464)

[IV. Xây dựng chương trình 51](#_Toc54651465)

[1. Thư viện và cộng cụ xây dựng ứng dụng. 51](#_Toc54651466)

[2. Thiết kế chương trình 51](#_Toc54651467)

[3. Hiện thực (Tương tác với API) giải thuật và thử nghiệm 52](#_Toc54651468)

[a. Tương tác với API 52](#_Toc54651469)

[b. Cấu trúc Project 53](#_Toc54651471)

[V. Kết luận 57](#_Toc54651472)

1. ***Giới thiệu đề tài***
2. **Tên đề tài và lý do chọn đề tài**

* *Tên đề tài:*

**“ Xây dựng kho dữ liệu và phát triển ứng dụng hỗ trợ tư vấn lương phù hợp với nghề nghiệp và trình độ ”**

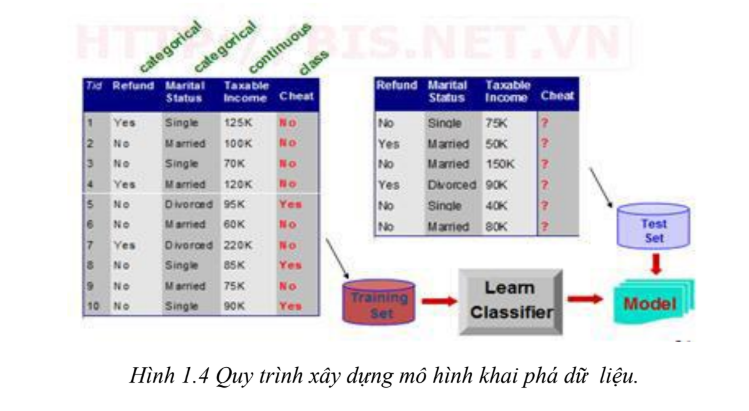
* *Lý do chọn đề tài:* Một trong những phần quan trọng nhất của buổi phỏng vấn xin việc là đàm phán lương. Tiền lương, thưởng và các khoản phụ cấp, chế độ hỗ trợ là vấn đề tương đối nhạy cảm với nhiều người, có thể khiến ứng viên cảm thấy "ngại" khi đề cập tới. Tuy nhiên, với những sinh viên mới ra trường như chúng em hoặc có thể đối với những người đã đi làm để xác định được mức lương tốt nhất cho vị trí công việc mới của bản than là việc hêt sức khó khăn và nan giải.

1. **Mục đích xây dựng đề tài**

* Mục đích nhóm chúng em xây dựng **“ứng dụng hỗ trợ tư vấn”** để giúp người dùng tự nhìn nhận và xác định giá trị của bản thân, xem trình độ học vấn, kỹ năng và kinh nghiệm làm việc đáng giá bao nhiêu trong thị trường việc làm hiện nay? Bằng cách này , người dùng sẽ có đầy đủ sự chuẩn bị để đàm phán mức lương có lợi cho mình. Nếu người dùng chưa biết cách giải quyết ra sao thì có thể tham khảo **qua “ứng dụng hỗ trợ tư vấn”** trong quá trình phỏng vấn dưới đây.

1. ***Xây dựng kho dữ liệu***
2. **Quy trình xây dựng mô hình khai phá dữ liệu:**

* Việc thực hiện một DMM với đầy đủ 4 bước công việc chính của quá trình khai phá dữ liệu là:
* **Bước 1**: Chuẩn bị dữ liệu (Data Preparation), trong bước này chúng ta thực hiện các công việc tiền xử lý dữ liệu theo yêu cầu của mô hình như trích chọn thuộc tính, rời rạc hóa dữ liệu và cuối cùng là chia dữ liệu nguồn (Data Source) thành 2 tập dữ liệu dùng để huấn luyện mô hình (Training Data) và kiểm tra mô hình (Testing data).
* **Bước 2**: Xây dựng mô hình (Data Modeling), ta sử dụng Training Data vừa tạo ra để xây dựng mô hình.
* **Bước 3**: Đánh giá mô hình (Validation), sau khi sử dụng Training Data để xây dựng mô hình, bây giờ ta sử dụng Testing Data để kiểm tra xem mô hình có đủ tốt để sử dụng hay không? (Nếu chưa đủ tốt thì phải sử dụng Training Data khác để huấn luyện lại). Có 3 kỹ thuật chính để kiểm tra mô hình đó là sử dụng Accuracy Chart (Lift Chart), Classification Matrix và Profit Chart.
* **Bước 4**: Sử dụng mô hình để dự đoán dự liệu trong tương lai (Model Usage), sau khi mô hình được kiểm tra (Testing) nếu độ chính xác đáp ứng yêu cầu thì có thể sử dụng model đã xây dựng vào dự đoán các dữ liệu chưa biết.



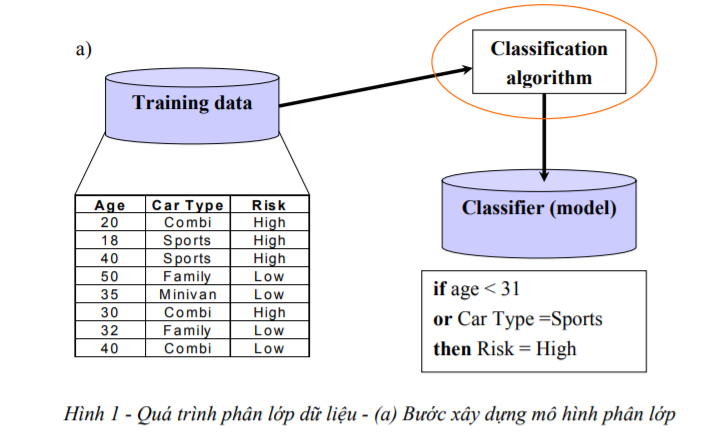
1. **Giới thiệu cấu hình quá trình bộ gom cụm phân lớp:**
2. *Giới thiệu về phân lớp:*

* **Mục đích**: Ngày nay phân lớp dữ liệu (classification) là một trong những hướng nghiên cứu chính của khai phá dữ liệu. Thực tế đặt ra nhu cầu là từ một cơ sở dữ liệu với nhiều thông tin ẩn con người có thể trích rút ra các quyết định nghiệp vụ thông minh, nhằm trích rút ra một mô hình mô tả các lớp dữ liệu quan trọng hay dự đoán xu hướng dữ liệu tương lai. Phân lớp dự đoán giá trị của những nhãn xác định (categorical label) hay những giá trị rời rạc (discrete value), có nghĩa là phân lớp thao tác với những đối tượng dữ liệu mà có bộ giá trị là biết trước.
* **Đầu vào**: một tập các mẫu dữ liệu huấn luyện, với một nhãn phân lớp cho mỗi mẫu dữ liệu
* **Đầu ra**: mô hình (bộ phân lớp) dựa trên tập huấn luyện và những nhãn phân lớp
* **Ví dụ**: mô hình phân lớp dự báo thời tiết có thể cho biết thời tiết ngày mai là mưa, hay nắng dựa vào những thông số về độ ẩm, sức gió, nhiệt độ,… của ngày hôm nay và các ngày trước đó. Hay nhờ các luật về xu hướng mua hàng của khách hàng trong siêu thị, các nhân viên kinh doanh có thể ra những quyết sách đúng đắn về lượng mặt hàng cũng như chủng loại bày bán…
* **Ứng dụng**: trong các lĩnh vực khác nhau như học máy (machine learning), hệ chuyên gia (expert system), thống kê (statistics), thương mại, nhà băng, maketing, nghiên cứu thị trường, bảo hiểm, y tế, giáo dục...

1. *Quy trình phân lớp:*

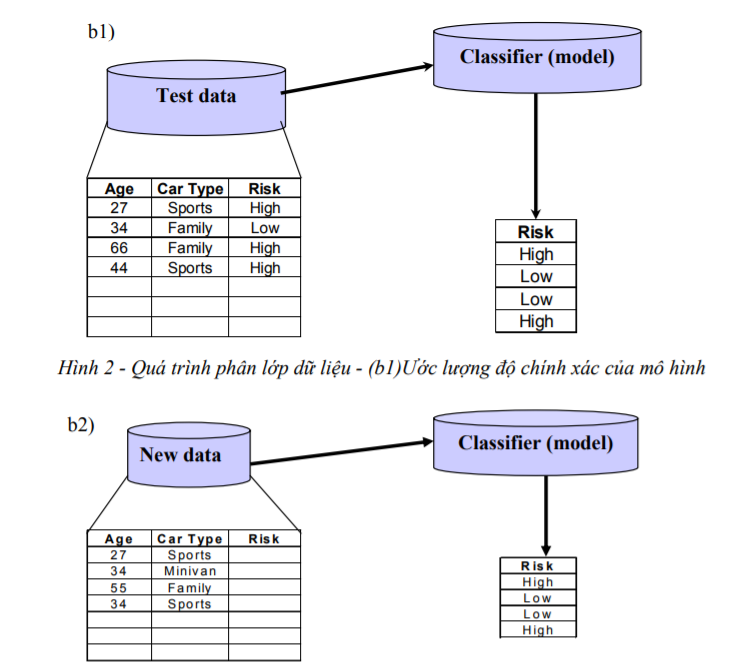
**Bước 1:** Xây dựng mô hình từ tập huấn luyện (learning).

* Mỗi bộ/mẫu dữ liệu được phân vào một lớp được xác định trước.
* Lớp của một bộ/mẫu dữ liệu được xác định bởi thuộc tính gán nhãn lớp
* Tập các bộ/mẫu dữ liệu huấn luyện - tập huấn luyện - được dùng để xây dựng mô hình
* Mô hình được biểu diễn bởi các luật phân lớp, các cây quyết định hoặc các công thức toán học



**Bước 2**: Sử dụng mô hình- kiểm tra tính đúng đắn của mô hình và dùng nó để phân lớp dữ liệu mới (classification)

* Đánh giá độ chính xác của mô hình:
* Lớp biết trước của một mẫu/bộ dữ liệu đem kiểm tra được so sánh với kết quả thu được từ mô hình
* Tỉ lệ chính xác = phần trăm các mẫu/bộ dữ liệu được phân lớp đúng bởi mô hình trong số các lần kiểm tra
* Phân lớp cho những đối tượng mới hoặc chưa được phân lớp

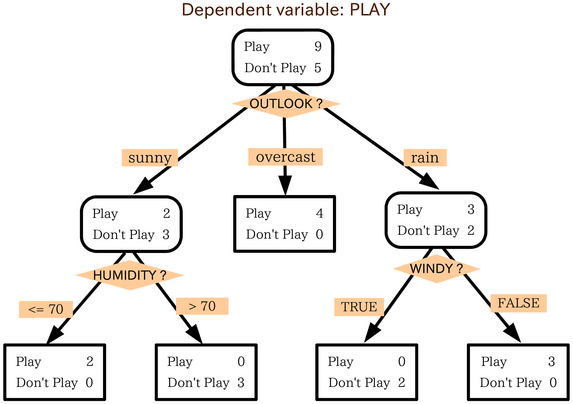


1. *Các kỹ thuật phân lớp:* 
   * Phân lớp cây quyết định (Decision tree classification)
   * Bộ phân lớp Bayesian (Bayesian classifier)
   * Mô hình phân lớp K-hàng xóm gần nhất (K-nearest neighbor classifier)
   * Mạng nơron
   * Phân tích thống kê
   * Các thuật toán di truyền
   * Phương pháp tập thô (Rough set Approach)
2. *Cây quyết định*

* Cây quyết định (Decision Tree) là một cây phân cấp có cấu trúc được dùng để phân lớp các đối tượng dựa vào dãy các luật (series of rules).Các thuộc tính của đối tượng (ngoại trừ thuộc tính phân lớp – Category attribute) có thể thuộc các kiểu dữ liệu khác nhau (Binary, Nominal, ordinal, quantitative values) trong khi đó thuộc tính phân lớp phải có kiểu dữ liệu là Binary hoặc Ordinal. Tóm lại, cho dữ liệu về các đối tượng gồm các thuộc tính cùng với lớp (classes) của nó, cây quyết định sẽ sinh ra các luật để dự đoán lớp của các đối tượng chưa biết (unseen data).
* Hai giai đoạn tạo cây quyết định:
* Xây dựng cây :
* Bắt đầu, tất cả các mẫu huấn luyện đều ở gốc o phân chia các mẫu dựa trên các thuộc tính được chọn
* Kiểm tra các thuộc tính được chọn dựa trên một độ đo thống kê hoặc heuristic
* Thu gọn cây : Xác định và loại bỏ những nhánh nhiễu hoặc tách khỏi nhóm
* Ví dụ:
* Đây là quản lý của một câu lạc bộ đánh golf nổi tiếng. Anh ta đang có rắc rối chuyện các thành viên đến hay không đến. Có ngày ai cũng muốn chơi golf nhưng số nhân viên câu lạc bộ lại không đủ phục vụ. Có hôm, không hiểu vì lý do gì mà chẳng ai đến chơi, và câu lạc bộ lại thừa nhân viên.
* Mục tiêu của David là tối ưu hóa số nhân viên phục vụ mỗi ngày bằng cách dựa theo thông tin dự báo thời tiết để đoán xem khi nào người ta sẽ đến chơi golf. Để thực hiện điều đó, anh cần hiểu được tại sao khách hàng quyết định chơi và tìm hiểu xem có cách giải thích nào cho việc đó hay không.
* Vậy là trong hai tuần, anh ta thu thập thông tin về: Trời (*outlook*) (nắng (*sunny*), và tất nhiên là số người đến chơi golf vào hôm đó. David thu được một bộ dữ liệu gồm 14 dòng và 5 cột.



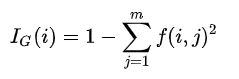
* Sau đó, để giải quyết bài toán của David, người ta đã đưa ra một mô hình cây quyết định.



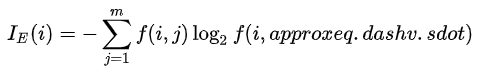
* Nhóm người chơi golf khi trời nắng, nhóm chơi khi trời nhiều mây, và nhóm chơi khi trời mưa.
* **Kết luận**:
* *Kết luận thứ nhất*: nếu trời nhiều mây, người ta luôn luôn chơi golf. Và có một số người ham mê đến mức chơi golf cả khi trời mưa.
* *Kết luận thứ hai*: ta lại chia nhóm trời nắng thành hai nhóm con. Ta thấy rằng khách hàng không muốn chơi golf nếu độ ẩm lên quá 70%.
* *Kết luận thứ ba*: ta chia nhóm trời mưa thành hai và thấy rằng khách hàng sẽ không chơi golf nếu trời nhiều gió.
* Và đây là lời giải ngắn gọn cho bài toán mô tả bởi cây phân loại. David cho phần lớn nhân viên nghỉ vào những ngày trời nắng và ẩm, hoặc những ngày mưa gió. Vì hầu như sẽ chẳng có ai chơi golf trong những ngày đó. Vào những hôm khác, khi nhiều người sẽ đến chơi golf, anh ta có thể thuê thêm nhân viên thời vụ để phụ giúp công việc.
* *Kết luận là cây quyết định giúp ta biến một biểu diễn dữ liệu phức tạp thành một cấu trúc đơn giản hơn rất nhiều.Có rất nhiều thuật toán phân lớp như ID3, J48, C4.5, CART (Classification and Regression Tree),... Việc chọn thuật toán nào để có hiệu quả phân lớp cao tùy thuộc vào rất nhiều yếu tố, trong đó cấu trúc dữ liệu ảnh hưởng rất lớn đến kết quả của các thuật toán. Chẳng hạn như thuật toán ID3 và CART cho hiệu quả phân lớp rất cao đối với các trường dữ liệu số (quantitative value) trong khi đó các thuật toán như J48, C4.5 có hiệu quả hơn đối với các dữ liệu Qualititive value (ordinal, Binary, nominal).*

1. *Các công thức:*

* Gini impurity:
* Dùng trong thuật toán CART (Classification and Regression Trees). Nó dựa vào việc bình phương các xác suất thành viên cho mỗi thể loại đích trong nút. Giá trị của nó tiến đến cực tiểu (bằng 0) khi mọi trường hợp trong nút rơi vào một thể loại đích duy nhất.
* Giả sử *y* nhận các giá trị trong {1, 2,..., *m*} và gọi *f(i,j)* là tần suất của giá trị *j* trong nút *i*. Nghĩa là *f(i,j)* là tỷ lệ các bản ghi với *y*=*j* được xếp vào nhóm *i*.



* Entropy
* Dùng trong các thuật toán sinh cây [ID3](https://vi.wikipedia.org/w/index.php?title=Thu%E1%BA%ADt_to%C3%A1n_ID3&action=edit&redlink=1), [C4.5](https://vi.wikipedia.org/w/index.php?title=Thu%E1%BA%ADt_to%C3%A1n_C4.5&action=edit&redlink=1) và C5.0. Số đo này dựa trên khái niệm [entropy](https://vi.wikipedia.org/wiki/Entropy) trong [lý thuyết thông tin](https://vi.wikipedia.org/wiki/L%C3%BD_thuy%E1%BA%BFt_th%C3%B4ng_tin) (*information theory*).



1. **Đặc tả dữ liệu:**

* Xây dựng kho dữ liệu và phát triển ứng dụng hỗ trợ tư vấn lương phù hợp với nghề nghiệp và trình độ với các thuộc tính đầu vào như sau:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên Trường** | **Mô Tả** |
| 1 | Kỹ Năng Mềm | Thuộc tính này cho biết kỹ năng mềm của nhân viên như thuyết trình, làm việc nhóm,… |
| 2 | Ngôn Ngữ Lập Trình | Thuộc tính này cho biết ngôn ngữ lập trình mà nhân viên sử dụng |
| 3 | Quy Mô Công Ty | Thuộc tính này cho biết quy mô của công ty mà nhân viên làm |
| 4 | Lĩnh Vực | Thuộc tính này cho biết lĩnh vực công việc mà nhân viên làm |
| 5 | Khu Vực | Thuộc tính này cho biết về vị trí địa lý của công ty |
| 6 | Ngoại Ngữ | Thuộc tính này cho biết kỹ năng ngoại ngữ khác mà nhân viên thành thạo |
| 7 | Kinh Nghiệm | Thuộc tính này cho biết cấp bậc được phân chia dựa trên số năm làm việc đã từng làm của nhân viên liên quan đến vị trí ứng tuyển hiện tại |
| 8 | Lương | Thuộc tính này cho biết lương mà Nhân Viên được hưởng |

|  |  |
| --- | --- |
| **STT** | **Giá Trị** |
| 1 | GT |
| 2 | QLTG |
| 3 | TT |
| 4 | LVN |

* *Kỹ năng mềm:*
* GT: Giao tiếp
* QLTG: Quản lý thời gian
* TT: Thuyết trình
* LVN: Làm việc nhóm

|  |  |
| --- | --- |
| **STT** | **Giá Trị** |
| 1 | LON |
| 2 | TB |
| 3 | NHO |

* *Quy mô công ty*:

|  |  |
| --- | --- |
| **STT** | **Giá Trị** |
| 1 | BackEnd |
| 2 | FrontEnd |
| 3 | Mobile |
| 4 | BackEnd,FrontEnd |
| 5 | BackEnd,Mobile |
| 6 | FrontEnd,Mobile |
| 7 | Mobile,BackEnd,FrontEnd |

* *Lĩnh vực*:

|  |  |
| --- | --- |
| **STT** | **Giá Trị** |
| 1 | KV1 |
| 2 | KV2 |
| 3 | KV3 |

* *Khu vực*:
* KV1: Thành phố lớn
* KV2: Thành phố đang phát triển
* KV3: Thành phố mới

|  |  |
| --- | --- |
| **STT** | **Giá Trị** |
| 1 | PB |
| 2 | TB |
| 3 | KPB |

* *Ngôn ngữ lập trình*:
* PB: Phổ biến
* TB: Trung bình
* KPB: Không phổ biến

|  |  |
| --- | --- |
| **STT** | **Giá Trị** |
| 1 | TT |
| 2 | CB |
| 3 | KC |
| 4 | DV |

* *Ngoại ngữ*:
* TT: Thành thạo
* CB: Cơ bản
* DV: Đọc viết
* KC: Không có

|  |  |
| --- | --- |
| **STT** | **Giá Trị** |
| 1 | Internship |
| 2 | Fresher |
| 3 | Junior |
| 4 | Senior |

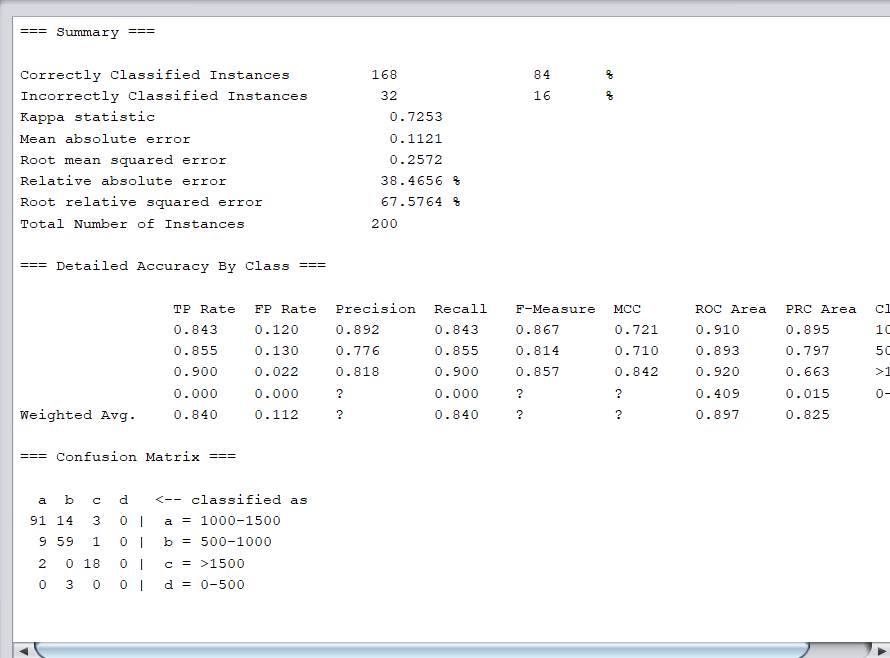
* *Kinh nghiệm*:

|  |  |
| --- | --- |
| **STT** | **Giá Trị** |
| 1 | 0-500 |
| 2 | 500-1000 |
| 3 | 1000-1500 |
| 4 | >1500 |

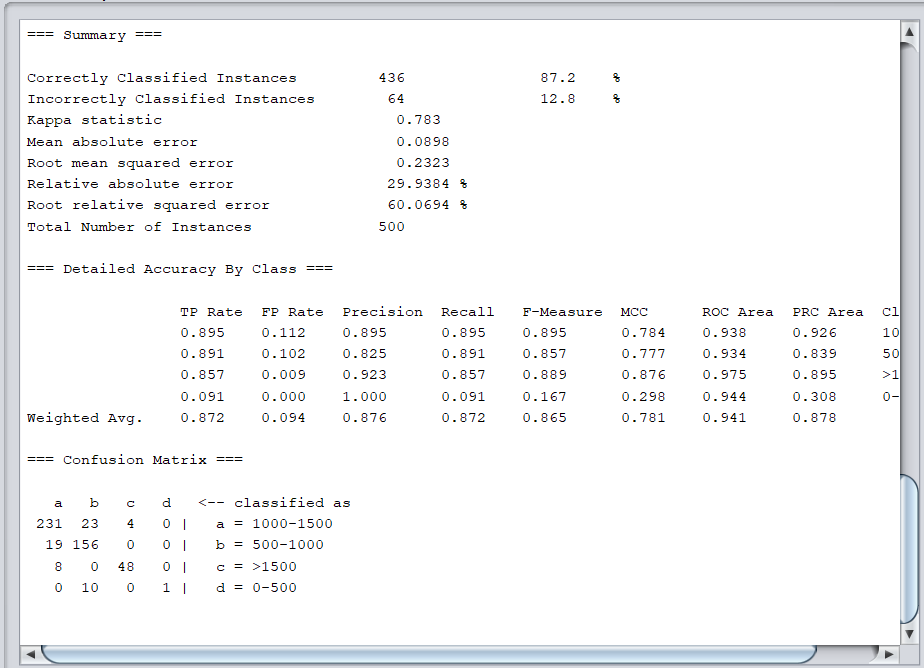
* *Lương (class)*:

1. **Sử dụng Weka để chọn được bộ dữ liệu ( Ổn định dữ liệu )**

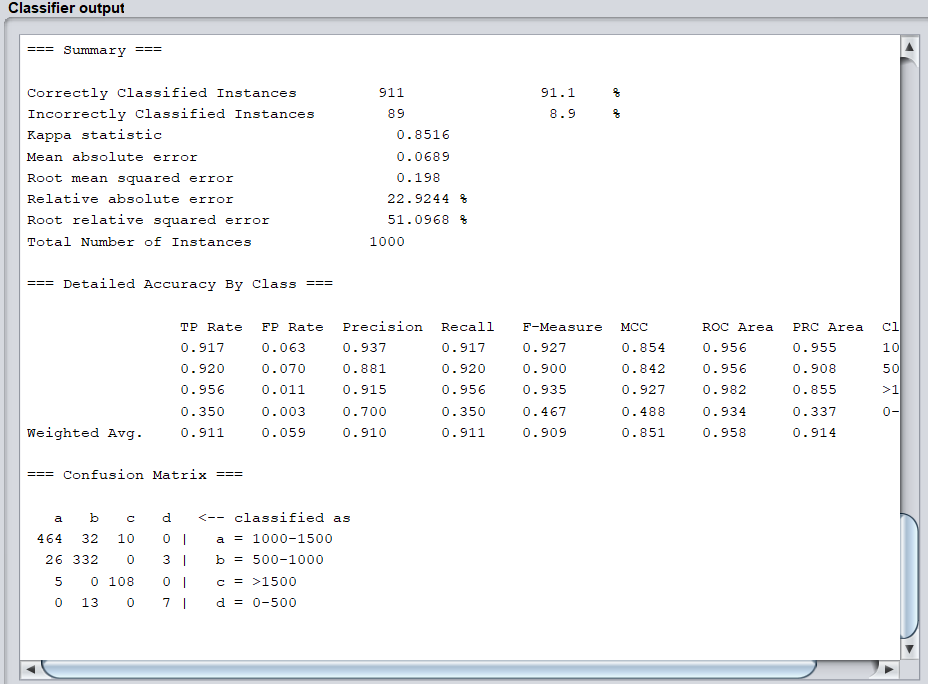
* Dữ liệu gồm 200 dòng:



* ***Nhận xét:*** Qua hình trên ta thấy khi chạy 200 mẫu tin chạy có 168 mẫu tin đúng ( chiếm 84%) và 32 mẫu tin sai ( chiếm 16%).
* Dữ liệu gồm 500 dòng:



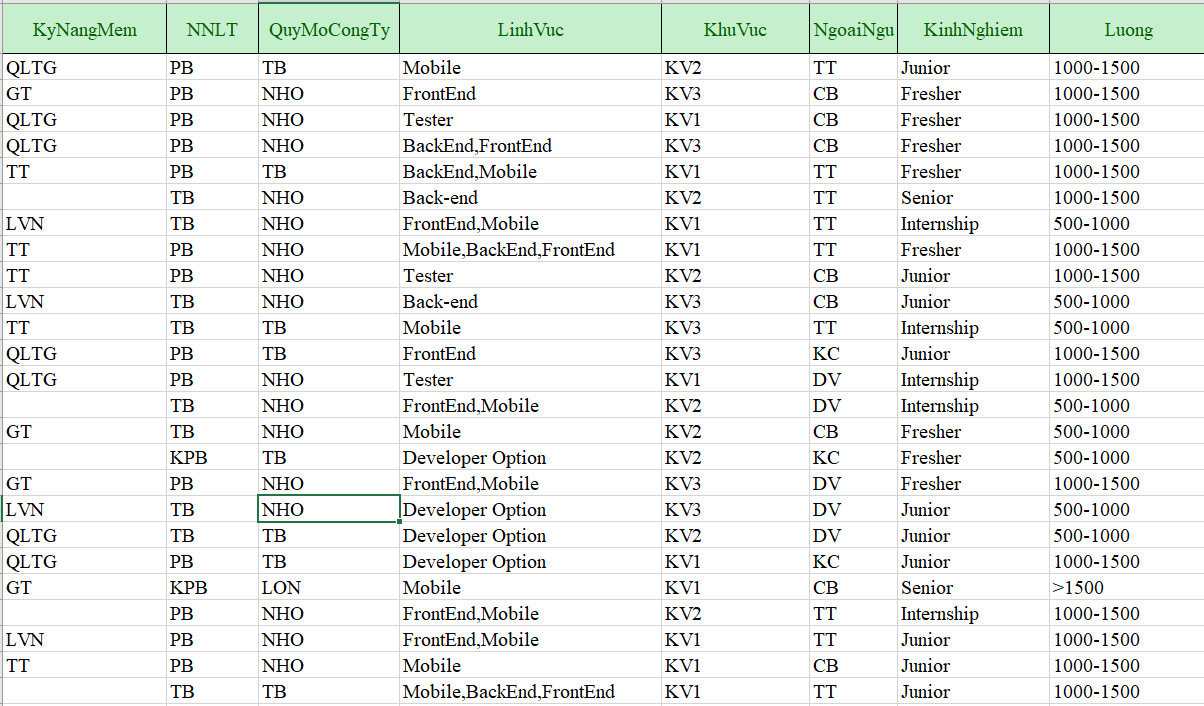
* ***Nhận xét:*** Qua hình trên ta thấy khi chạy 500 mẫu tin chạy có 436 mẫu tin đúng ( chiếm 87.2%) và 64 mẫu tin sai ( chiếm 12.8%).
* Dữ liệu gồm 1000 dòng



* ***Nhận xét:*** Qua hình trên ta thấy khi chạy 1000 mẫu tin chạy có 911 mẫu tin đúng ( chiếm 91.1%) và 89 mẫu tin sai ( chiếm 8.9%).
* ***Kết luận:*** *Dựa trên kết quả khi chạy 3 bộ dữ liệu với số dòng khác nhau cho kết quả mẫu tin đúng ở bộ dữ liệu 1000 dòng đạt kết quả tốt nhất. Chọn bộ dữ liệu 1000 dòng làm bộ dữ liệu gốc.*

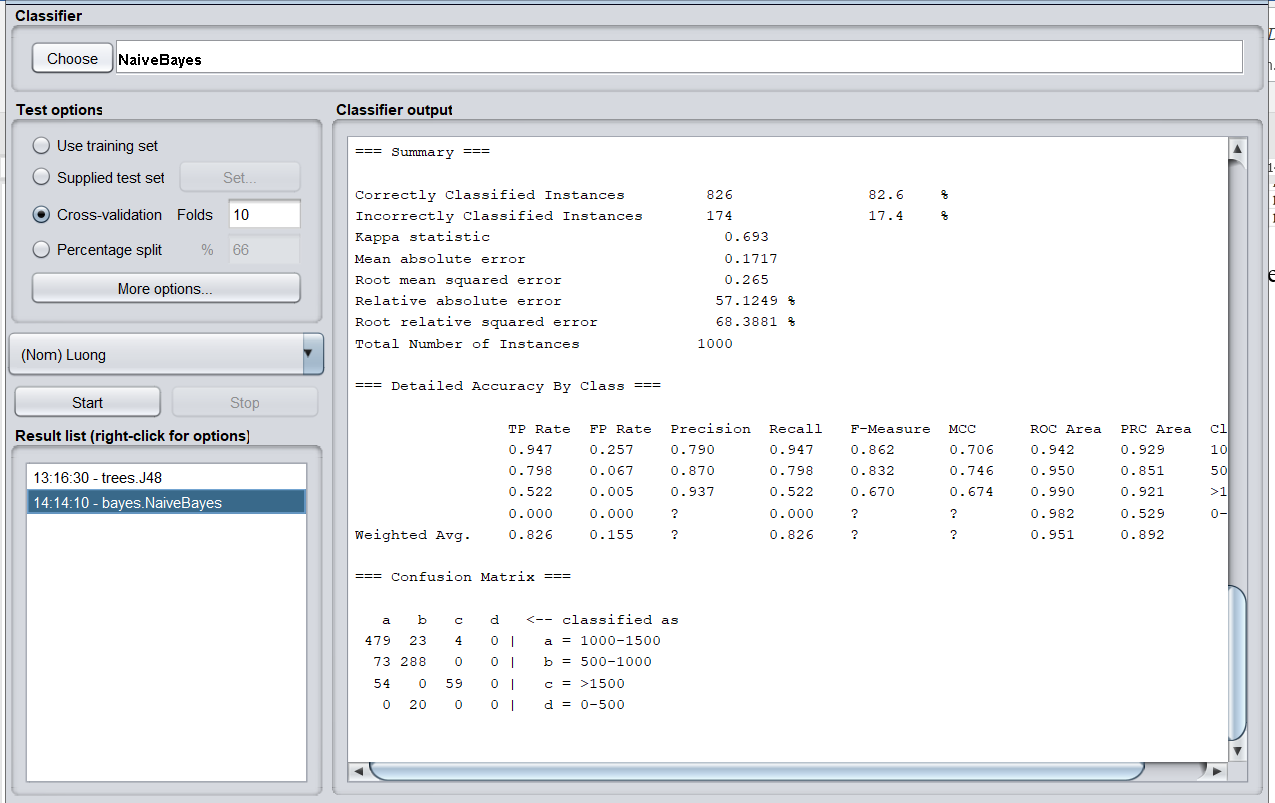
1. **Bộ dữ liệu gốc:**

* Bộ dữ liệu gồm 1000 dòng dữ liệu, được lưu trong file: ***DataMini4.csv***



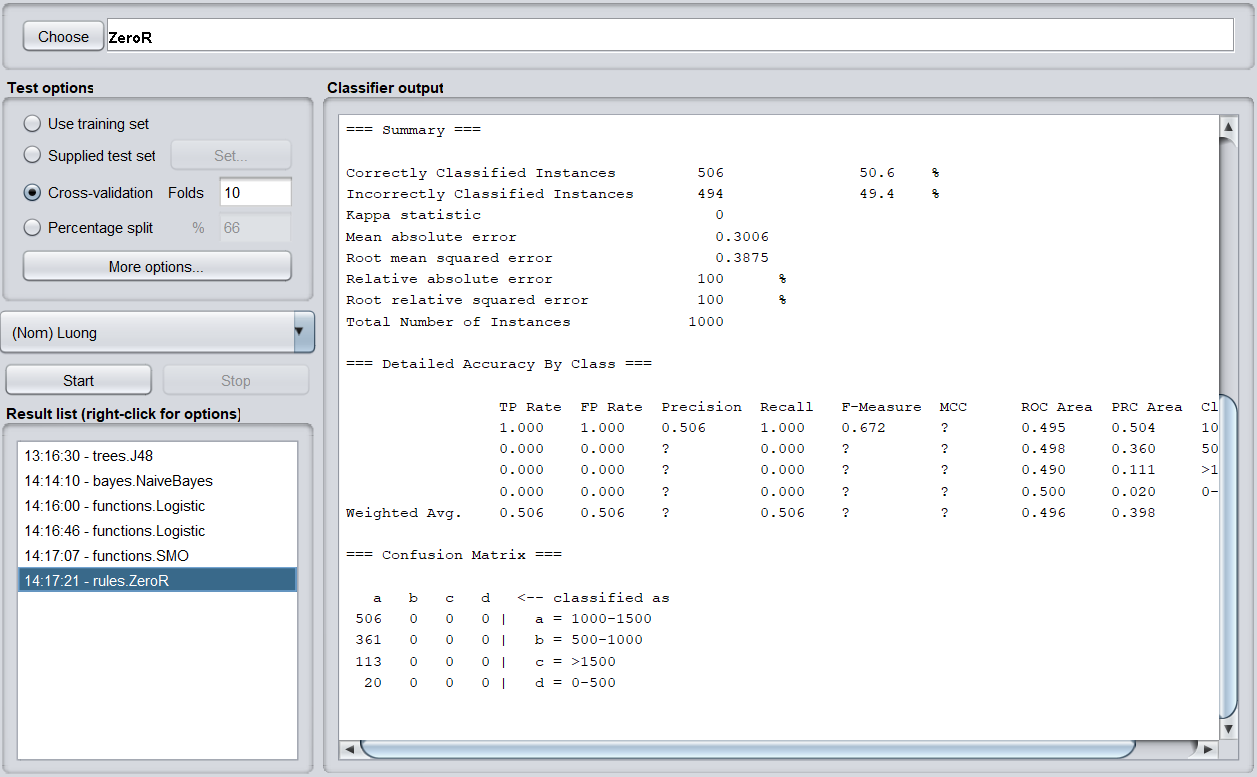
* ***Nhận xét:*** *Dữ liệu phù hợp cho việc phân tích cho việc hỗ trợ tư vấn lương theo nghề nghiệp và trình độ và khớp với đặc tả*

1. **Tiến hành xây dựng và đánh giá mô hình dữ liệu**
2. *Giải thuật NaiveBayes*



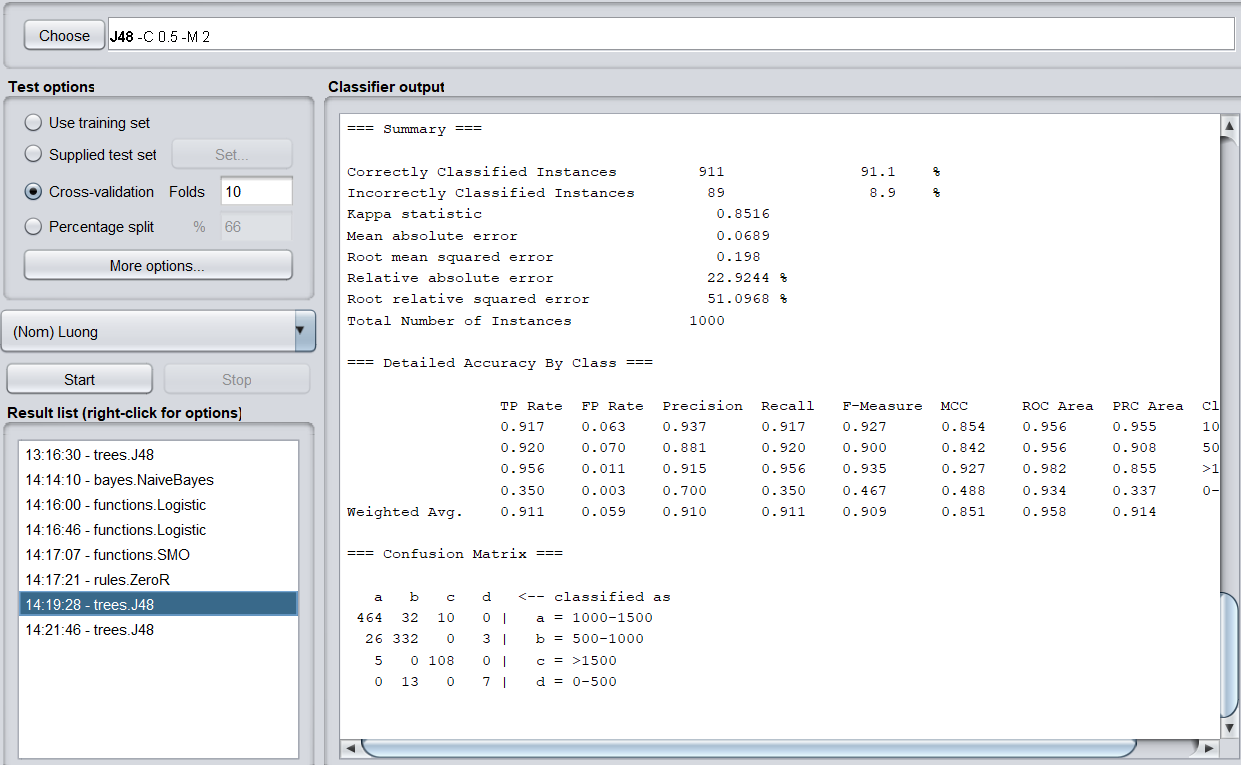
* ***Nhận xét:*** *Qua hình trên ta thấy khi chạy 1000 mẫu tin chạy có 826 mẫu tin đúng ( chiếm 82.6%) và 174 mẫu tin sai ( chiếm 17.4%).*

1. *Giải thuật ZeroR*



* ***Nhận xét:*** *Qua hình trên ta thấy khi chạy 1000 mẫu tin chạy có 506 mẫu tin đúng ( chiếm 50.6%) và 494 mẫu tin sai ( chiếm 49.4%).*

1. *Giải thuật J48 với độ tin cậy là 0.25:*



* ***Nhận xét:*** *Qua hình trên ta thấy khi chạy 1000 mẫu tin chạy có 911 mẫu tin đúng ( chiếm 91.1%) và 89 mẫu tin sai ( chiếm 8.9%).*

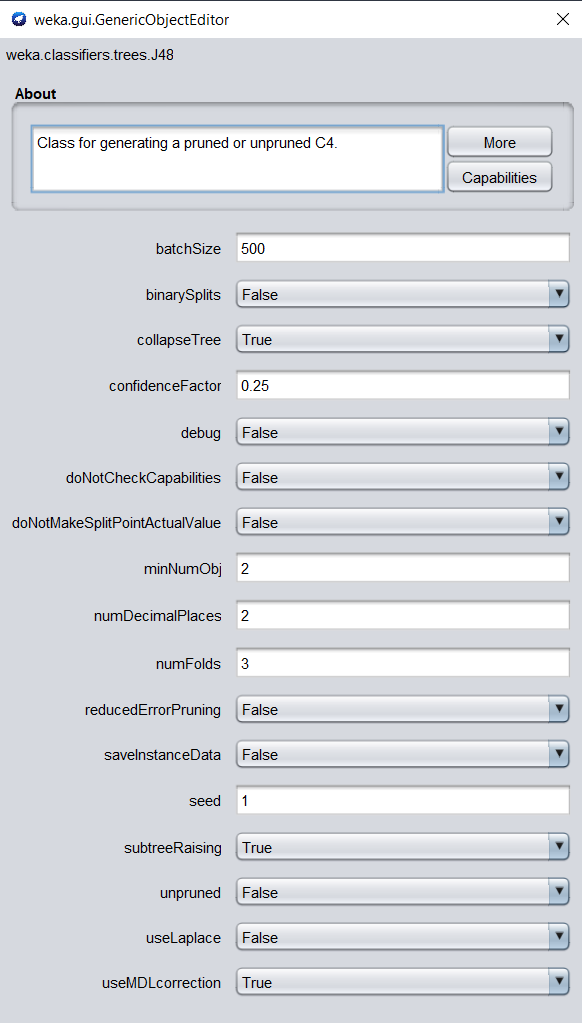
1. *Giải thuật J48 với độ tin cậy là 0.5:*



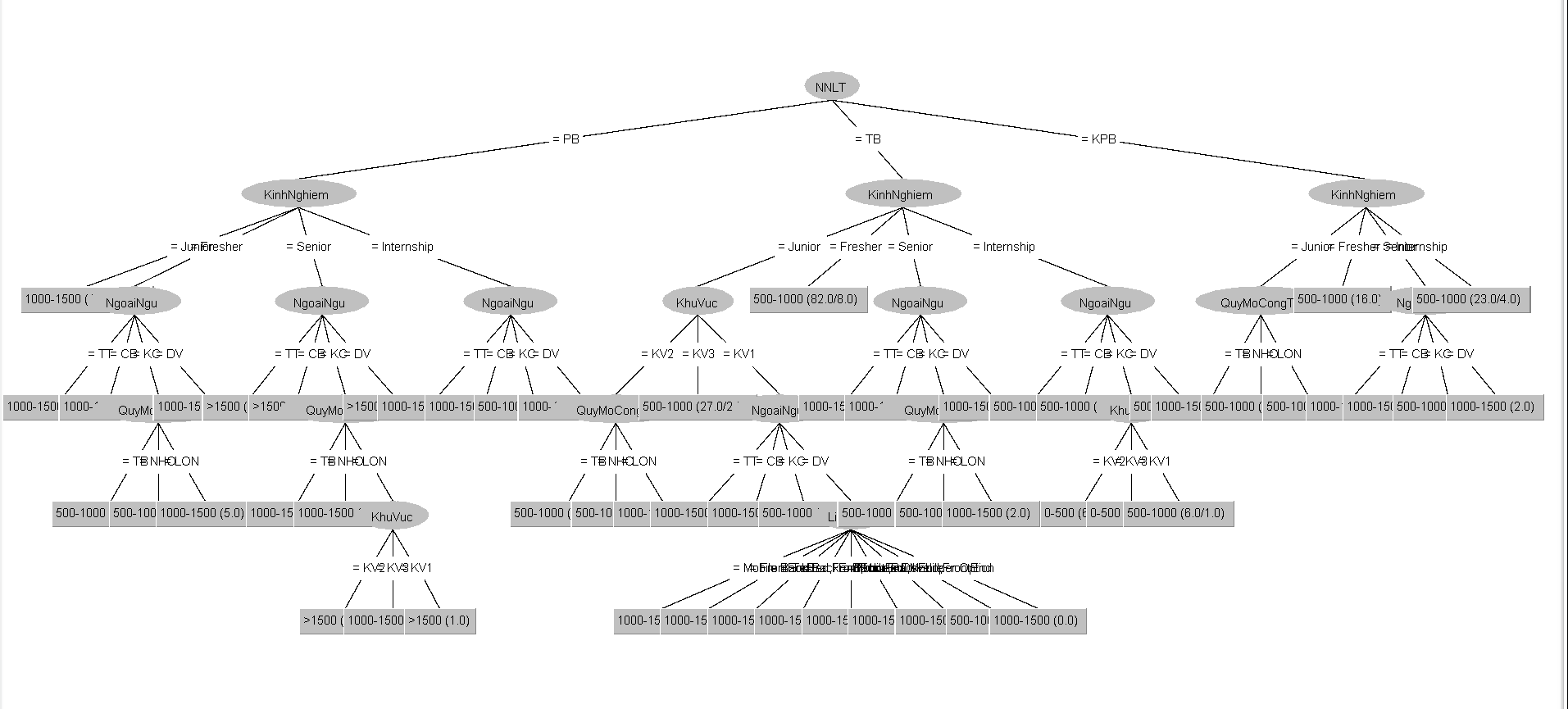
* ***Nhận xét:*** *Qua hình trên ta thấy khi chạy 1000 mẫu tin chạy có 910 mẫu tin đúng ( chiếm 91%) và 90 mẫu tin sai ( chiếm 9%).*
* ***Kết luận:*** *Từ 1 số giải thuật trên thuật toán J48 với độ tin cậy là 0.25 khá cao và cây quyết định ra được kết quả khá giống với dự tính ban đầu của nhóm.*

1. **Chọn cấu hình chạy giải thuật:**

* **BatchSize** (Bó dữ liệu)là số lượng mẫu mà ta lấy để huấn luyện, ở đây ta chọn **500** vì số lượng mẫu có trong data là 1000 mẫu nên ta được 2 bó. **ConfidenceFactor** là mức độ tin cậy vào phương pháp, ở đây ta chọn **0.25** vì đây là data mà ta tự chuẩn bị chứ không phải thực tế nên có thể tin cậy được. **Unpruned** ( Không cắt tỉa cây) ở đây ta chọn false để chương trình tự động cắt tỉa cây quyết định.



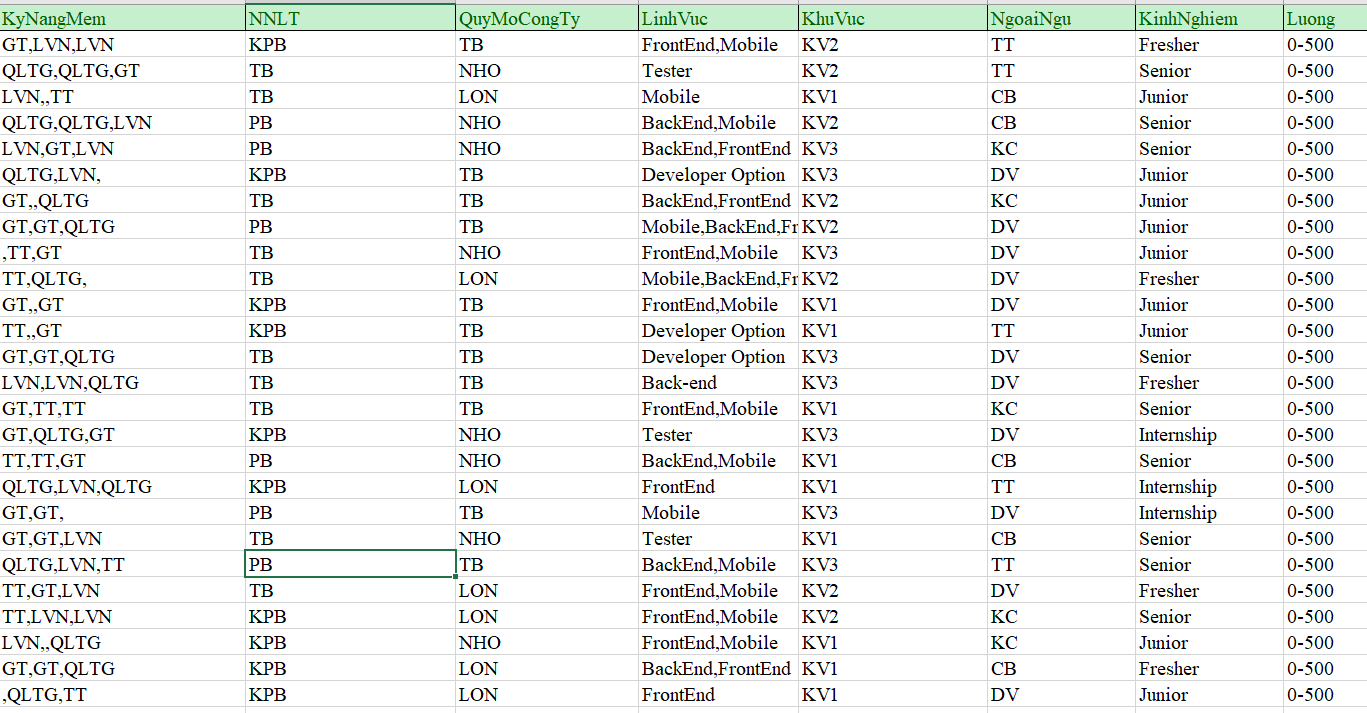
* Cây quyết định với cấu hình trên:



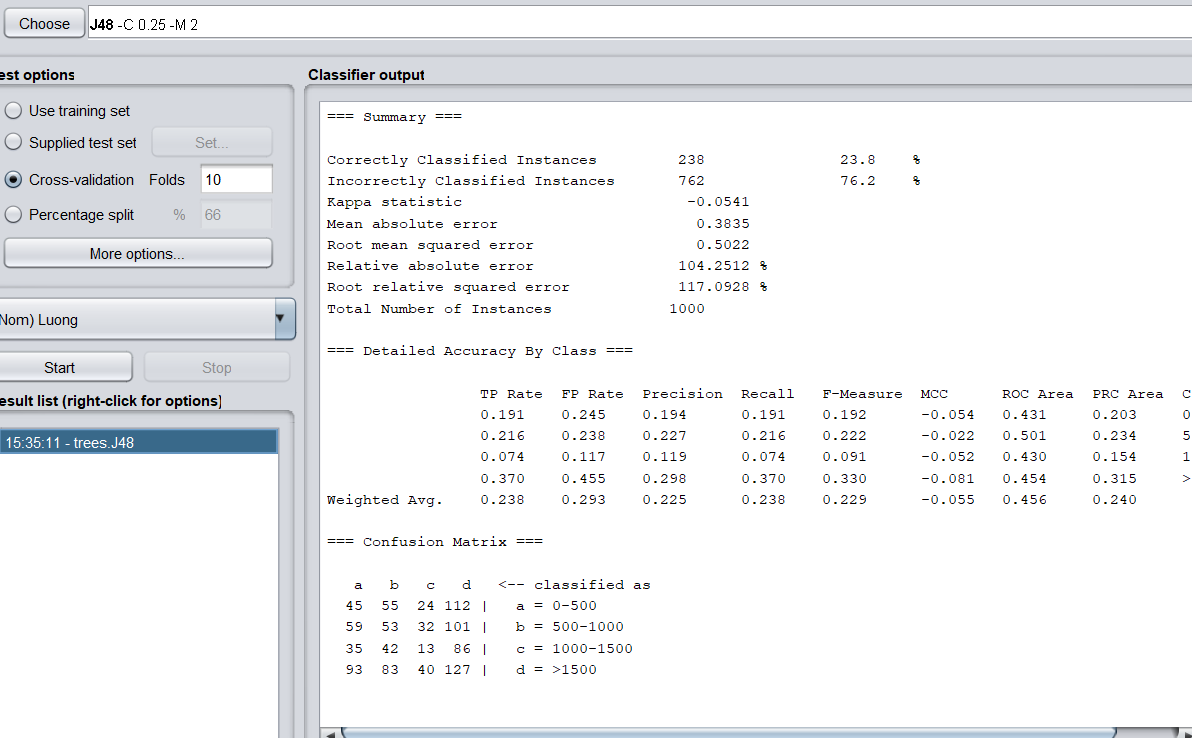
* Dữ liệu dự đoán lương dựa trên mức độ phổ biến của ngôn ngữ lập trình.
* Nếu ngôn ngữ lập trình PB xét theo kinh nghiệm:
* Senior tùy theo trình độ ngoại ngữ sẽ có mức lương tương ứng là trên >1500 $. Nếu không có ngoại ngữ sẽ xét theo quy mô công ty. Nếu quy mô công ty NHO hoặc TB thì mức lương từ 1000-1500$. Nếu quy mô công ty lớn sẽ xét tiếp theo khu vực. Nếu là KV3 sẽ có mức lương từ 1000-1500$, nếu là KV1 hoặc KV2 mức lương trên 1500$.
* Junior mức lương là 1000-1500
* Fresher xét theo trình độ ngoại ngữ.Nếu có ngoại ngữ sẽ được mức lương từ 1000-1500. Nếu không có ngoại ngữ sẽ xét theo quy mô công ty. Với quy mô TB hoặc NHO mức lương là 500-1000$, quy mô lớp mức lương 1000-1500$.
* Internship xét theo trình độ ngoại ngữ. Nếu không có ngoại ngữ mức lương là 500-1000$. Có ngoại ngữ mức lương là 1000-1500$.
* Nếu ngôn ngữ lập trình TB xét theo kinh nghiệm:
* Fresher mức lương từ 500-1000$
* Internship xét theo trình độ ngoại ngữ. Nếu có ngoại ngữ mức lương là 500-1000$, nếu không có ngoại ngữ sẽ xét theo khu vực. KV2 hoặc KV3 mức lương từ 0-500$, KV1 mức lương là 500-1000$.
* Senior xét theo trình độ ngoại ngữ. Nếu có ngoại ngữ mức lương là 1000-1500$, nếu không có ngoại ngữ sẽ xét theo quy mô công ty. Nếu quy mô công ty NHO hoặc TB thì mức lương từ 500-1000$, quy mô công ty LON mức lương 1000-1500$.
* Junior xét theo khu vực. KV3 mức lương từ 500-1000$. KV2 xét theo quy mô công ty. Nếu quy mô công ty NHO hoặc TB thì mức lương từ 500-1000$, quy mô công ty LON mức lương 1000-1500$. KV3 xét theo ngoại ngữ. Trình độ ngoại ngữ TT hoặc CB mức lương là 1000 – 1500$. Không có trình độ ngoại ngữ mức lương 500-1000$. Trình độ DV thì xét theo lĩnh vực ra mức lương tương ứng.
* Nếu ngôn ngữ lập trình KPB xét theo kinh nghiệm:
* Fresher mức lương từ 500-1000$
* Internship mức lương từ 500-1000$
* Junior xét theo quy mô công ty. Quy mô NHỎ hoặc LON mức lương là 500-1000$, quy mô TB thì mức lương 1000-1500$
* Senior xét theo trình độ ngoại ngữ. Nếu có ngoại ngữ mức lương là 1000-1500$, nếu không có ngoại ngữ mức lương là 500-1000$

1. **Tạo bộ dữ liệu thứ 2:**

* Bộ dữ liệu thứ 2 gồm 1000 dòng được lưu trong file: ***DataMini2.csv***



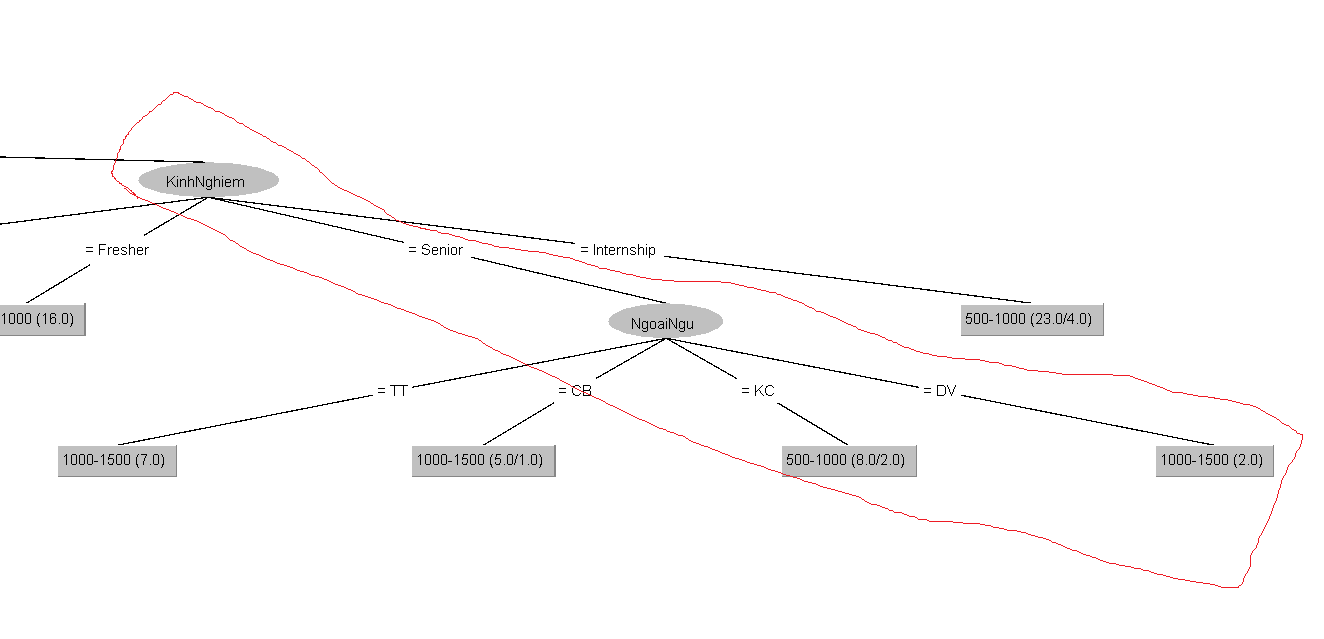
* Chạy trên giải thuật J48 với cấu hình giống bộ dữ liệu gốc.



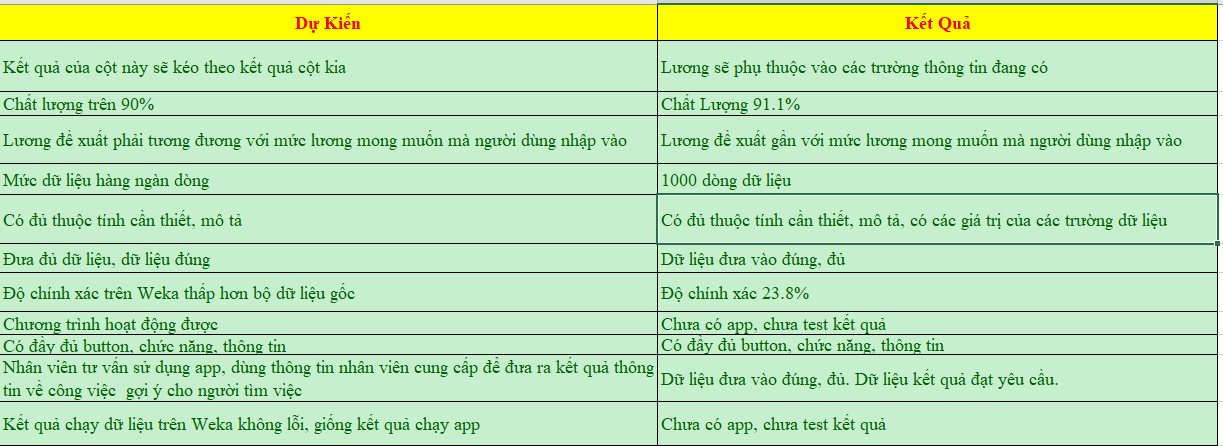
* ***Nhận xét:*** *Qua hình trên ta thấy khi chạy 1000 mẫu tin chạy có 238 mẫu tin đúng ( chiếm 23.8%) và 768 mẫu tin sai ( chiếm 76.8%).*
* ***Đánh giá*** *2 bộ dữ liệu thì kết quả cho thấy bộ dữ liệu gốc chính xác hơn và có thể tin tưởng.*

1. ***Test dữ liệu***
2. **Test dữ liệu lần thứ nhất:**

* Kết quả test dữ liệu cho thấy kết quả của thuộc này sẽ làm thay đổi kết quả của thuộc tính kia. Dẫn chứng bên dưới cho thấy. Cùng ngôn ngữ lập trình, cùng kinh nghiệm nhưng khác ngoại ngữ sẽ dẫn đến mức lương khác nhau.



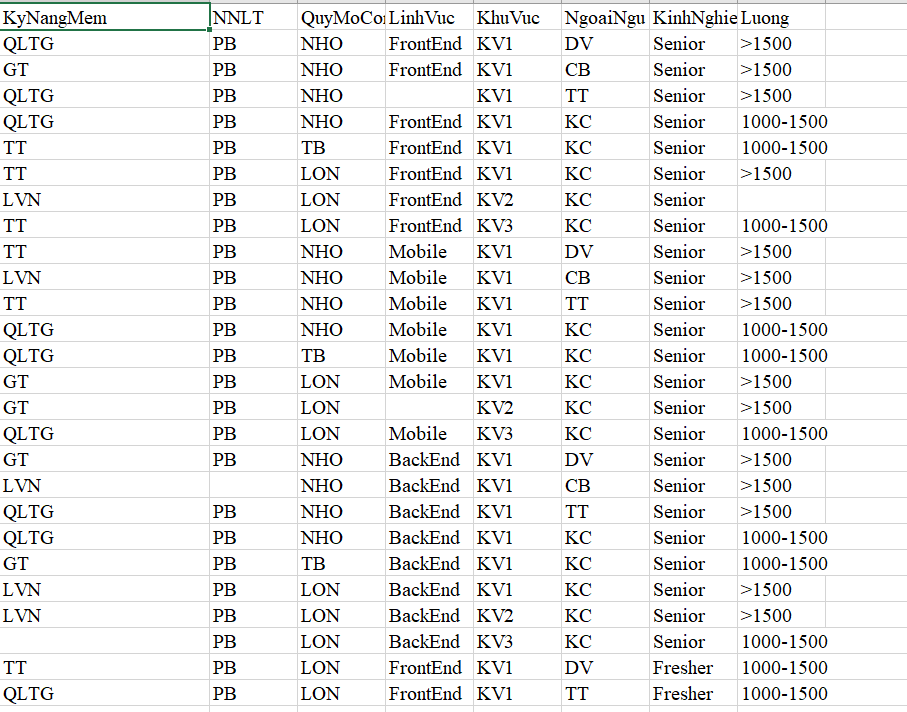
* Kiểm tra đặc tả dữ liệu NNLT (3 mức) pass
* Kỹ năng mềm (4 trường) pass
* Quy mô công ty (3 mức) pass
* Lĩnh vực (7 trường) pass
* Khu vực (3 trường) pass
* Ngoại ngữ (4 mức) pass
* Lương (4 mức) pass
* Kinh nghiệp (4 trường) pass
* *(Ghi chú : pass có đầy đủ thuộc tính đặc tả trong kho dữ liệu)*
* ***Kết quả test:***

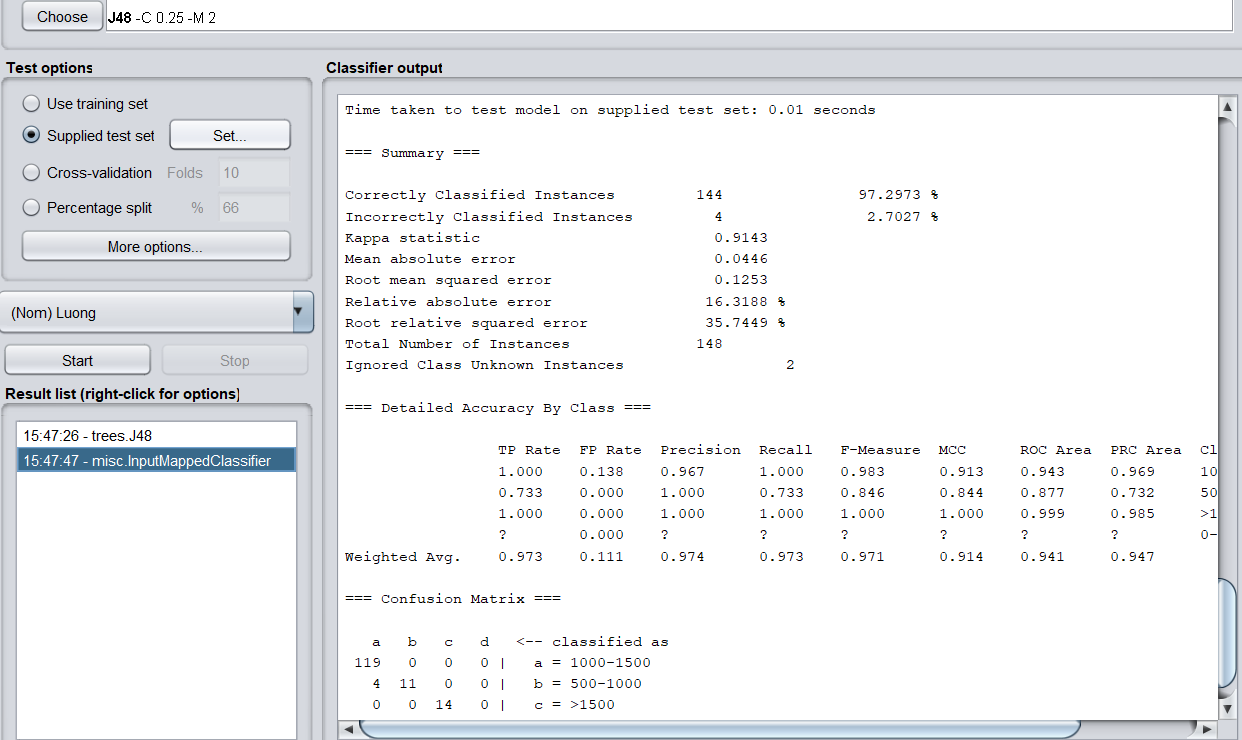


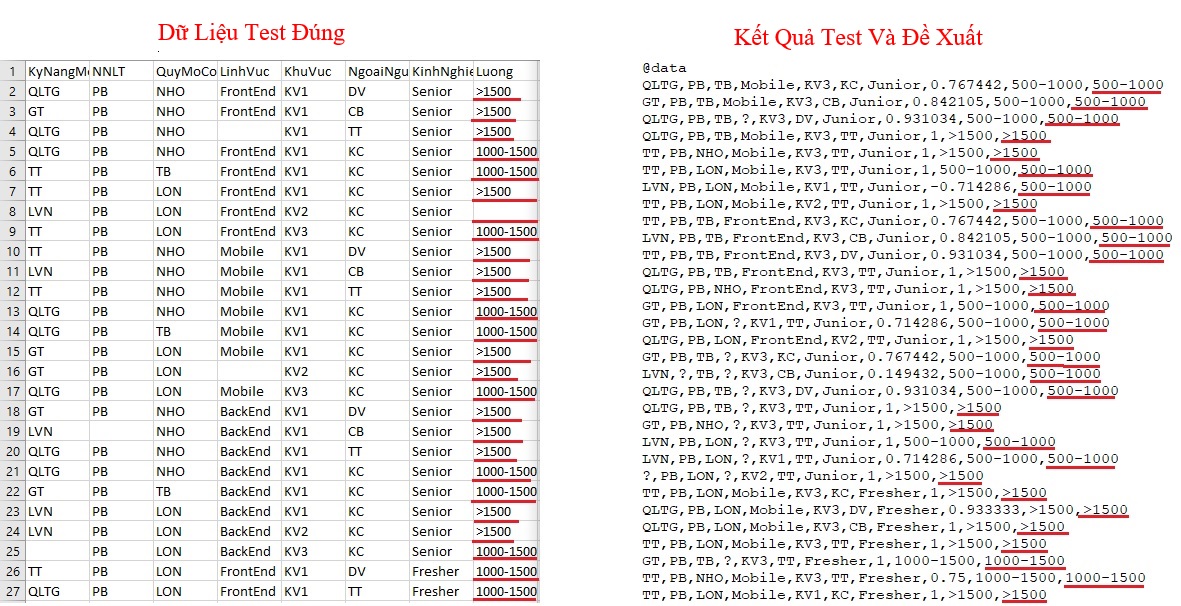
* ***Nhận xét:*** *Chất lượng bộ dữ liệu gốc khi chạy trên tập huấn luyện ( use training set) với thuật toán J48 và độ tin cậy 0.25 đạt độ chính xác là 93.1%.*

1. **Test dữ liệu lần thứ hai :**
2. *Tạo bộ dữ liệu test đúng với dữ liệu gốc*

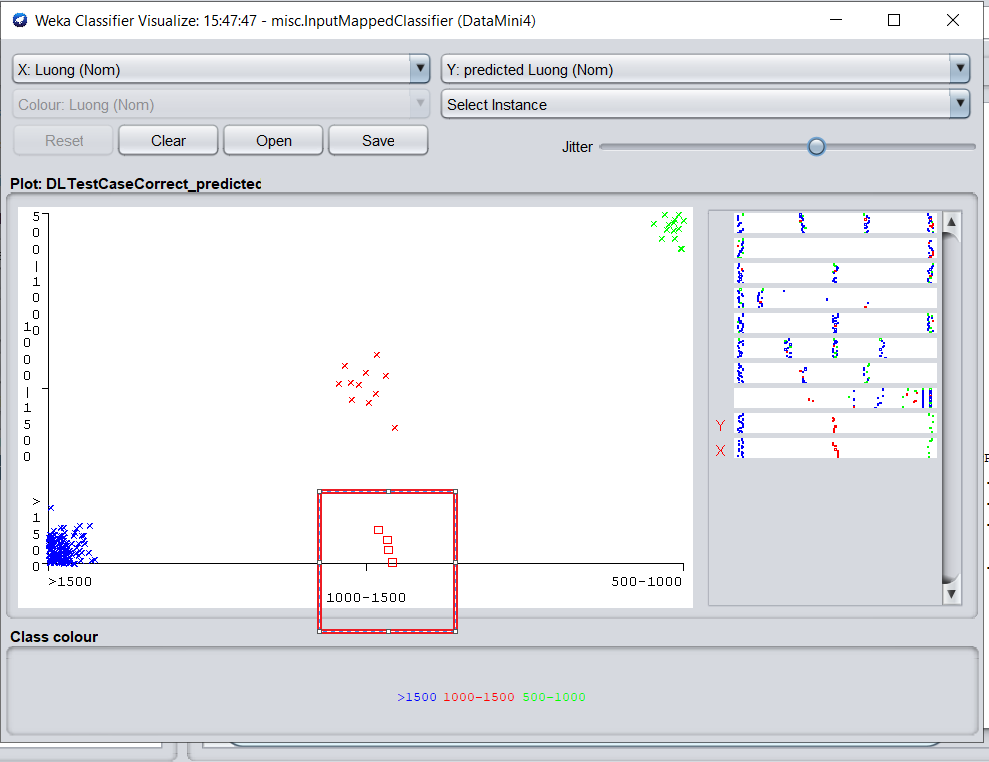
* Dữ liệu gồm 150 dòng được lưu ở file: ***DLTestCaseCorrect.csv***



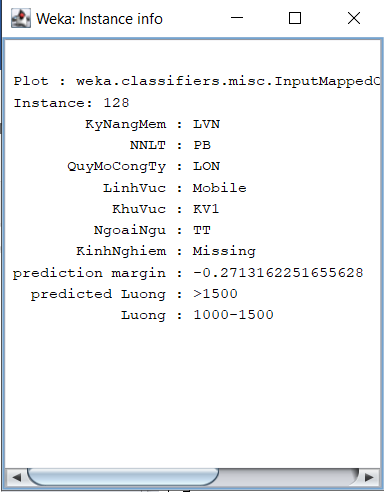




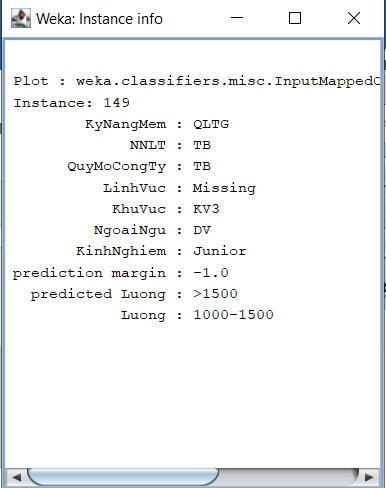
* Đối với dữ liệu sai các số liệu được thống kê ở trên dễ dàng thấy dữ liệu ở một số dòng gạch đỏ không khớp với nhau vậy đối với dữ liệu đúng thuật toán tự động sửa lại cho hợp lý với những gì nó học được và nó làm ảnh hưởng đến những đặc tả khác để đưa ra số liệu phù hợp nhất với trường hợp đó , sau đó mới đưa ra đề xuất đúng .



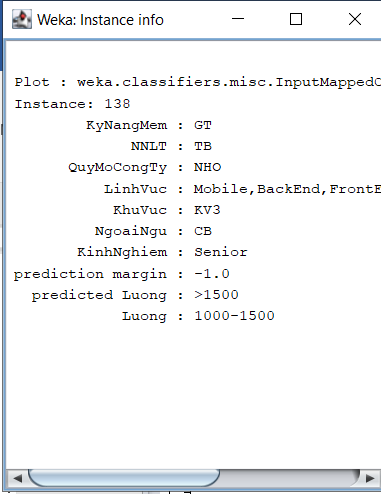
* Trong 93.1% đúng của bộ dữ liệu gốc có 97.29% đúng do bộ dữ liệu gốc không hoàn toàn chính xác. Kết quả test 150 dòng dữ liệu đúng có 4 trường hợp tô đỏ sai so với kết quả mong muốn. Lương mong muốn là 1000-1500 nhưng lương đề xuất là >1500 .
* Trên khung Visualize Classifier error cột X : Luong(Nom) biểu thị cho dữ liệu ở file test còn cột Y : predicted Luong(Nom) biểu thị cho tri thức khai phá học được từ file dữ liệu gốc . Đối với đề tài nó là tiền đề chuẩn để đưa ra mức lương đề xuất cho tất cả các file dữ liệu muốn test ( nghĩa là cột X ).
* Trường hợp sai tại dòng 128



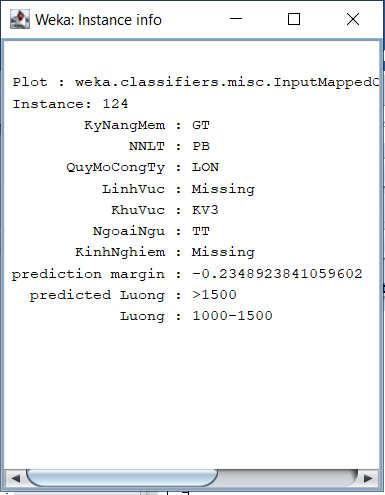
* Trường hợp sai tại dòng 149



* Trường hợp sai tại dòng 149

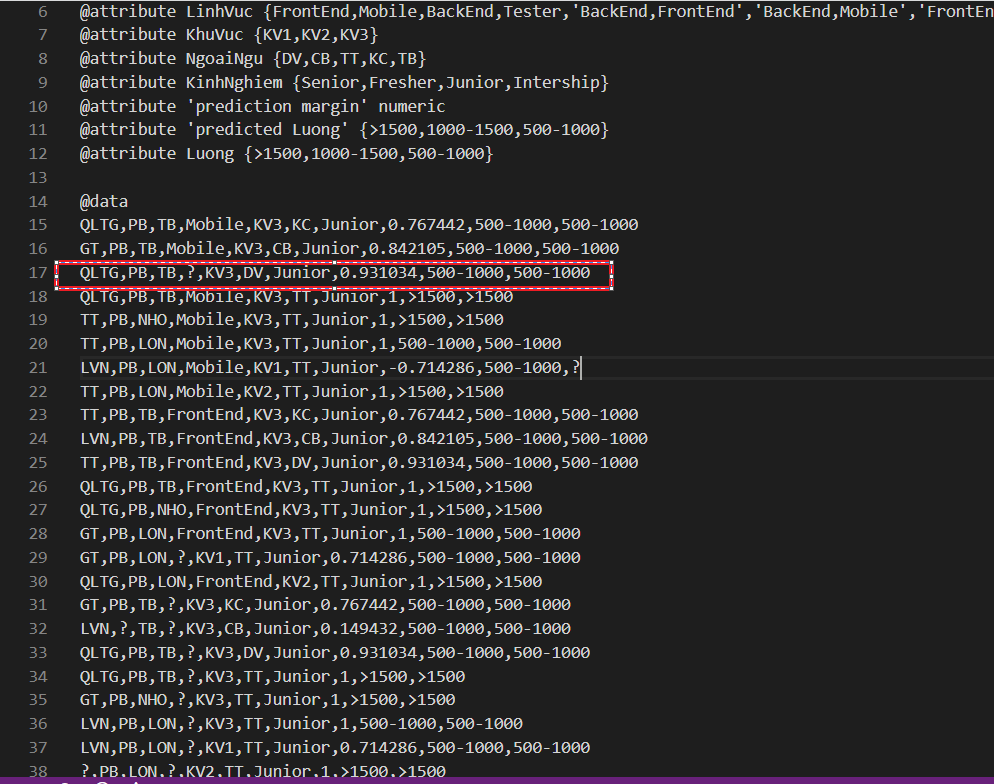


* Trường hợp sai tại dòng 124



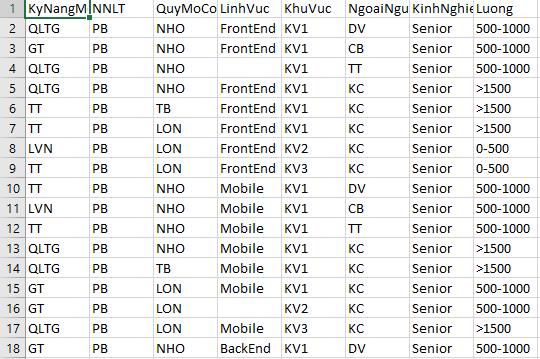
1. *Test dữ liệu khuyết:*

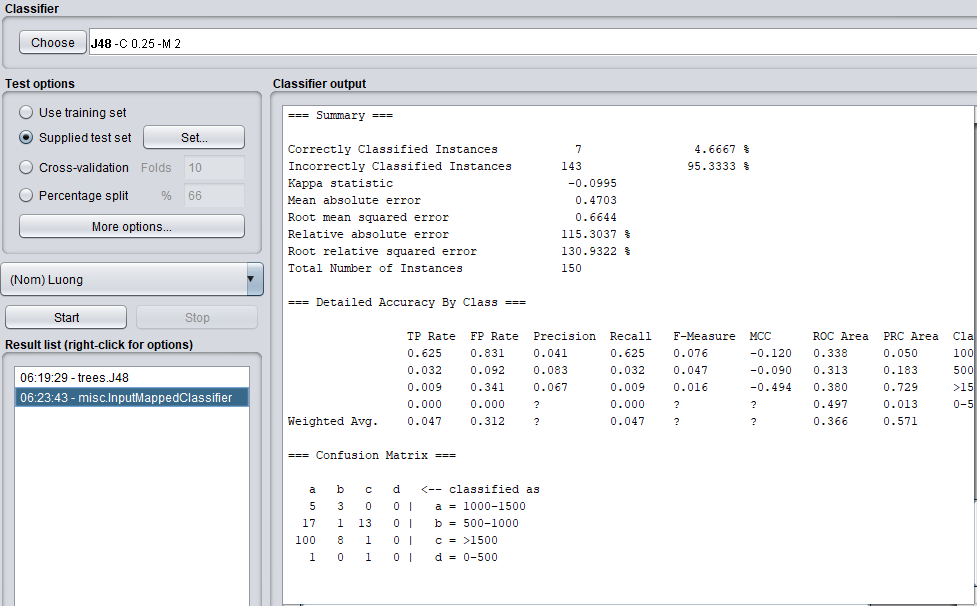
* Đối với trường hợp dữ liệu bị khuyết 1 đặc tả dữ liệu vẫn đưa ra chính xác về mặt ý nghĩa so với dữ liệu gốc và không gây ảnh hưởng đến những đặc tả khác (ví dụ người dùng không điền vào đặc tả trường ngoại ngữ thì chương trình sẽ dựa vào thông tin chuẩn ở các mục khác đề xuất ngược ra trình độ ngoại ngữ của họ). còn khuyết 2 trường hợp trở lên chương trình vẫn đưa ra đề xuất nhưng không thể đảm bảo chính xác hoàn toàn .



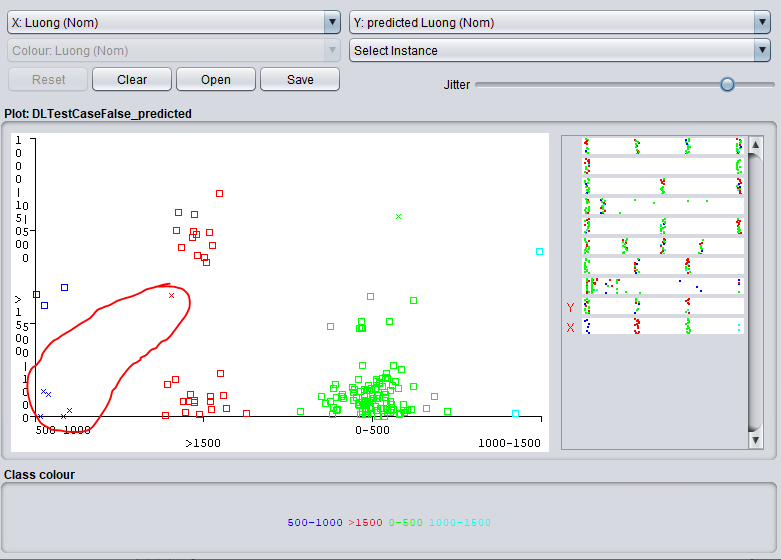
1. *Test bộ dữ liệu sai với dữ liệu gốc*

* Dữ liệu gồm 150 dòng được lưu ở file: ***DLTestCaseFalse.csv***

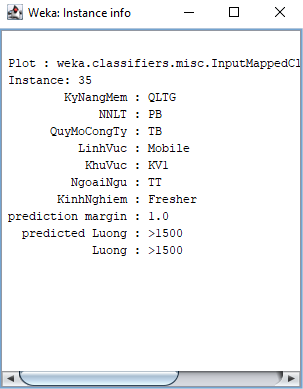




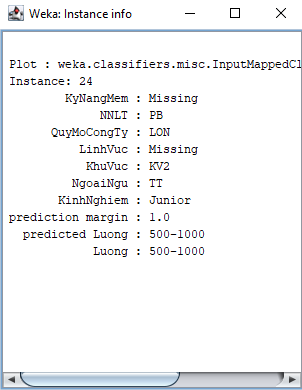
* Trong 93.1% đúng của bộ dữ liệu gốc có 4.6667% đúng so với bộ dữ liệu test . Kết quả test 150 dòng dữ liệu đúng có 6 trường hợp khoanh đỏ đúng so với kết quả mong muốn. Tất cả những mức lương sai trong bộ test sai đa số đều được đề xuất đúng với dữ liệu gốc mà nó học được.



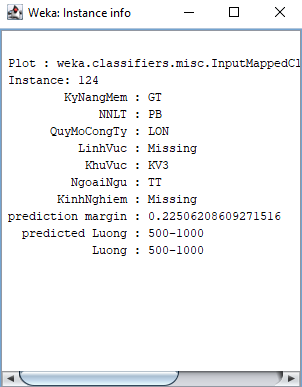
* Trường hợp đúng với dữ liệu sai dòng 35



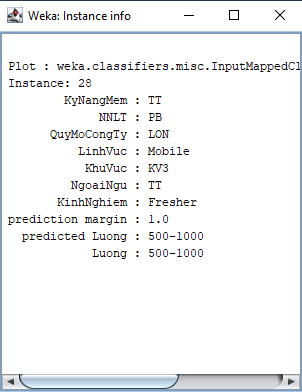
* Trường hợp đúng với dữ liệu sai dòng 24



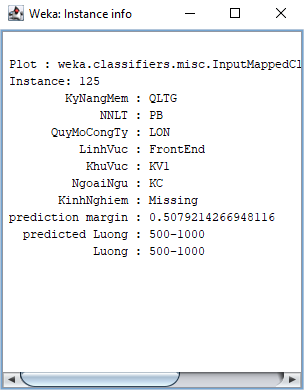
* Trường hợp đúng với dữ liệu sai dòng 124



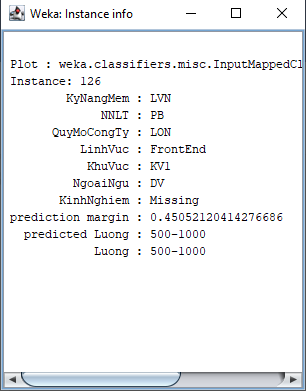
* Trường hợp đúng với dữ liệu sai dòng 28



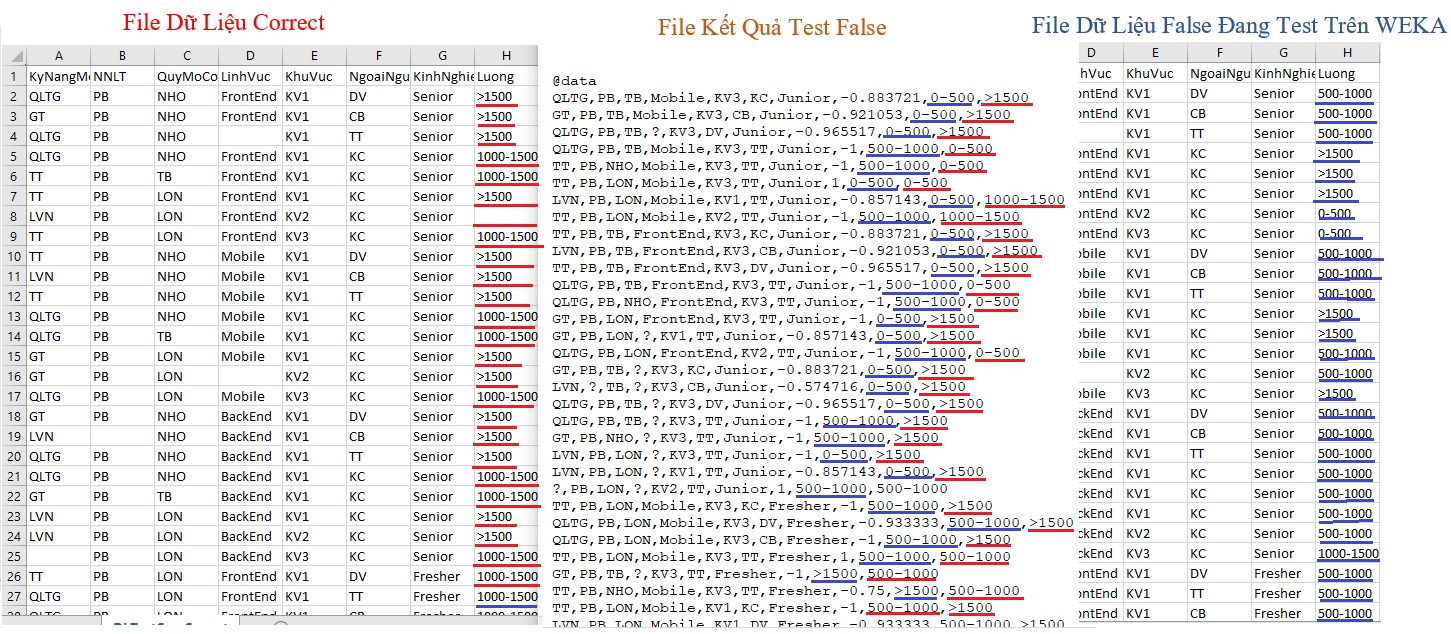
* Trường hợp đúng với dữ liệu sai dòng 125



* Trường hợp đúng với dữ liệu sai dòng 126



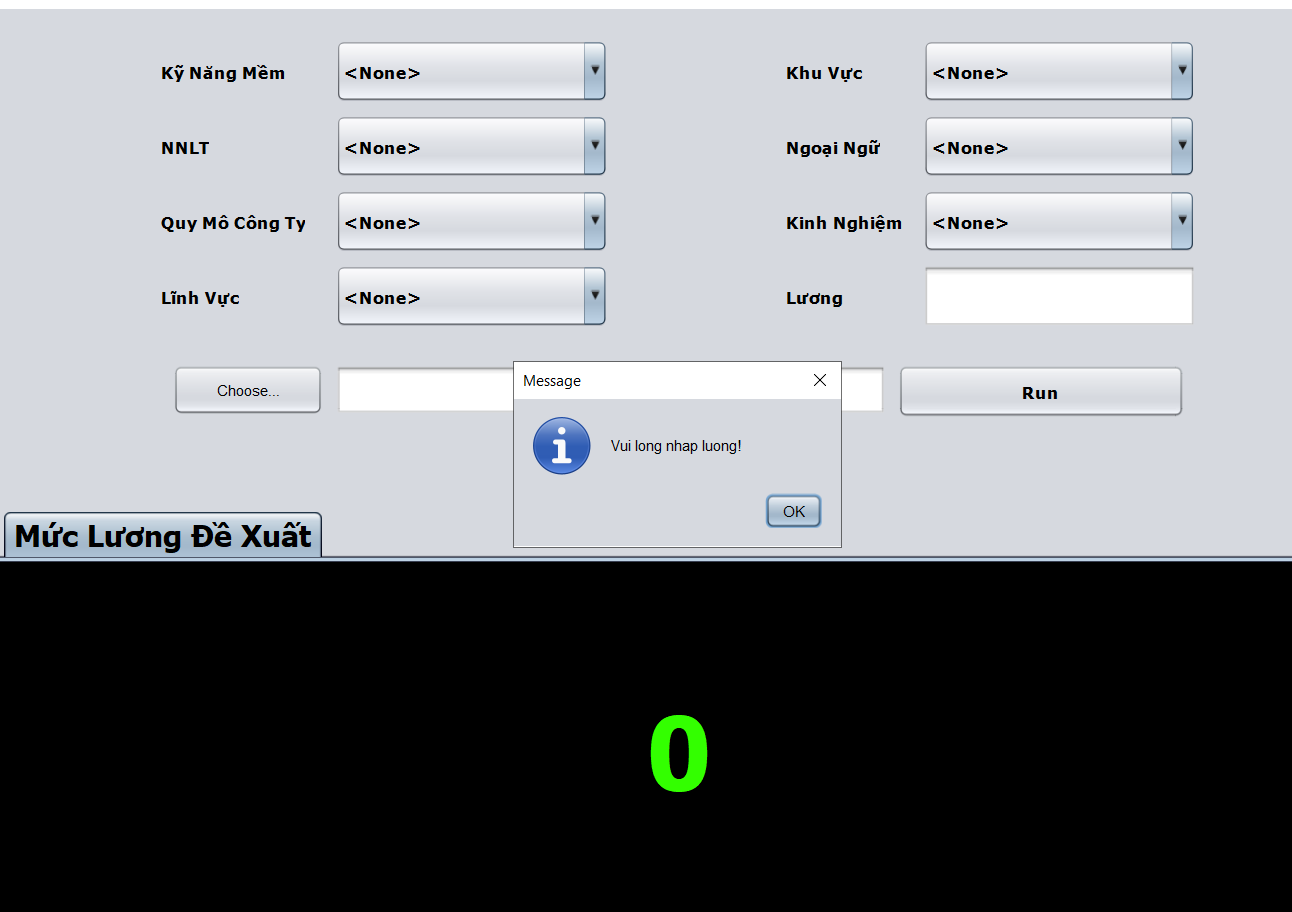
* So sánh kết quả với file dữ liệu đúng trên cơ sở của dữ liệu gốc ta dễ dàng thấy các gạch đỏ đa số đều khớp với nhau . còn gạch xanh là file dữ liệu test sai đang được test .



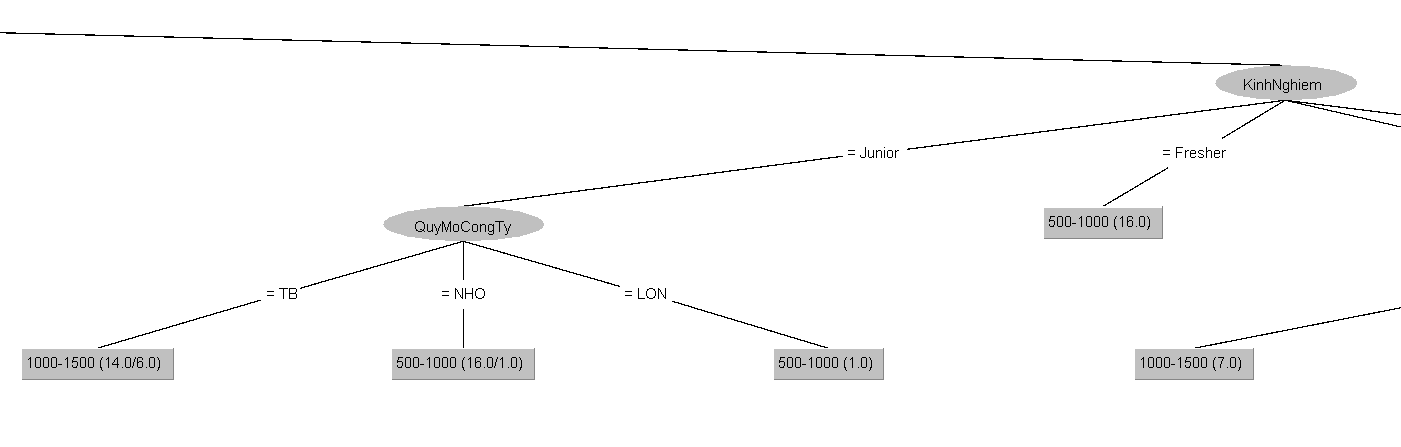
* So sánh kết quả với file dữ liệu đúng trên cơ sở của dữ liệu gốc ta dễ dàng thấy các gạch đỏ đa số đều khớp với nhau . còn gạch xanh là file dữ liệu test sai đang được test .
* Đối với dữ liệu sai các số liệu được thống kê ở trên dễ dàng thấy dữ liệu ở một số dòng gạch xanh không khớp với nhau vậy đối với dữ liệu sai thuật toán tự động sửa lại cho hợp lý với những gì nó học được và nó làm ảnh hưởng đến những đặc tả khác để đưa ra số liệu phù hợp nhất với trường hợp đó , sau đó mới đưa ra đề xuất đúng .

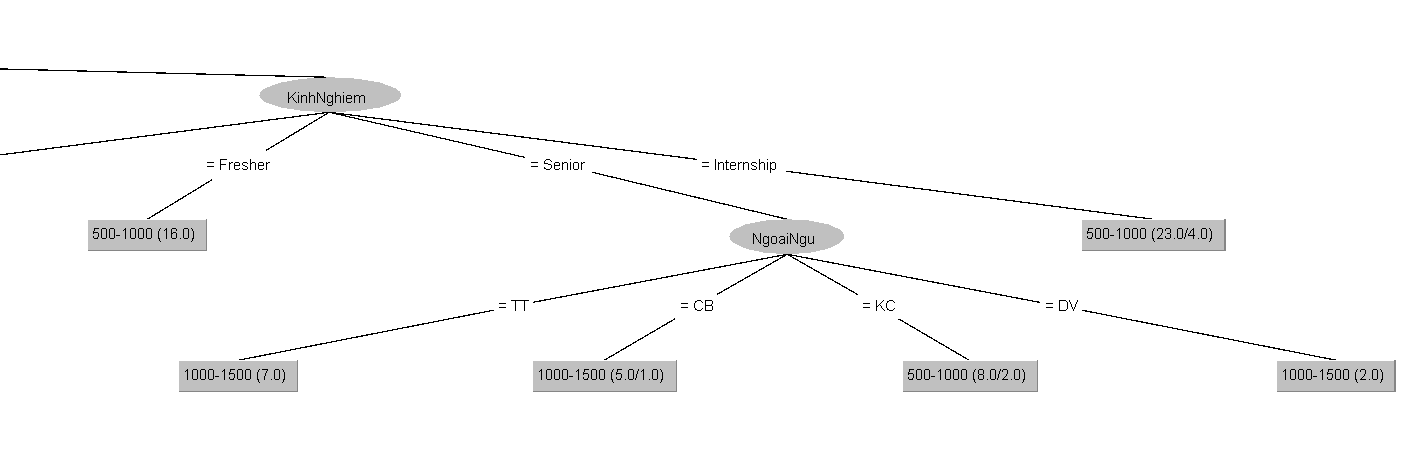
1. **Trên chương trình:**
2. *Chạy trên bộ dữ liệu gốc:*

* Những trường bắt buộc nhập dữ liệu như Lương hoặc Choose. Nếu không nhập sẽ có thông báo để người dùng nhập vào.

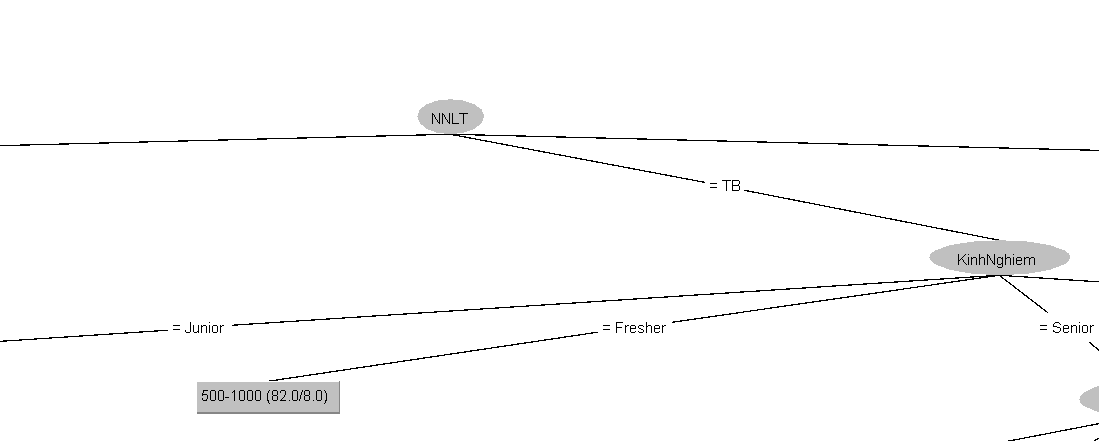


* Quá trình test trên chương trình ra kết quả giống như trên cây quyết định.

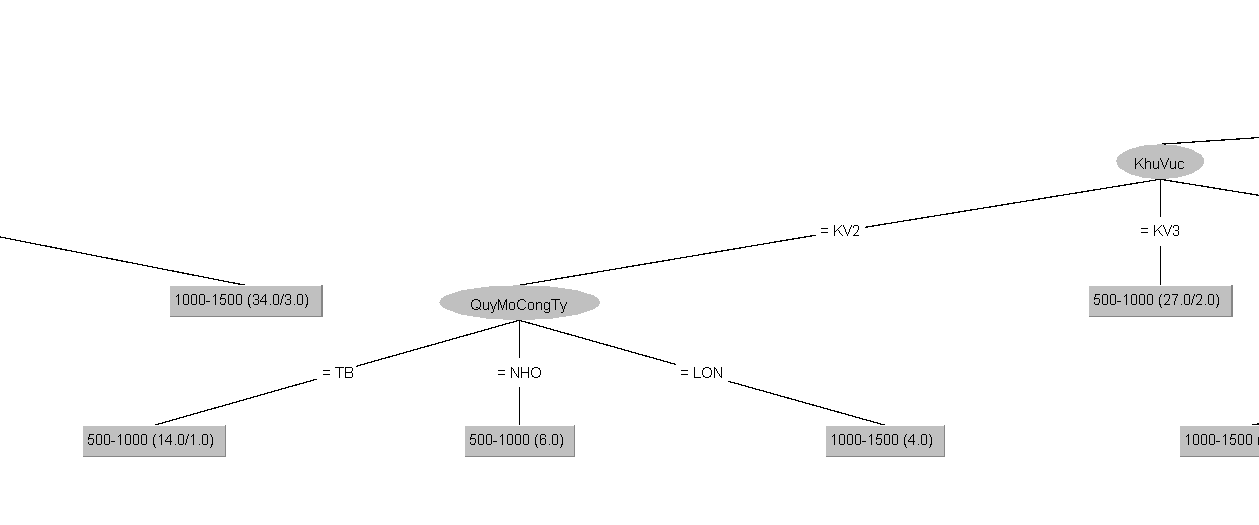




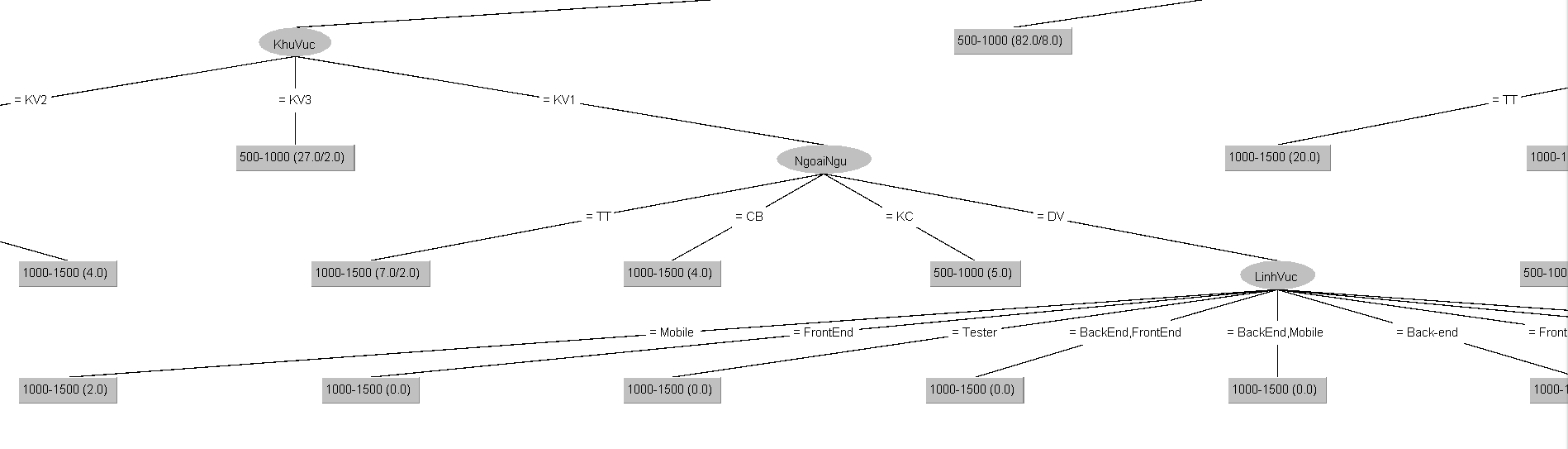
* Ngôn ngữ lập trình KPB => Kinh nghiệm Junior => Quy mô công ty: pass
* Ngôn ngữ lập trình KPB => Kinh nghiệm Fresher, Intership : pass
* Ngôn ngữ lập trình KPB => Kinh nghiệm Senior => Ngoại ngữ : pass



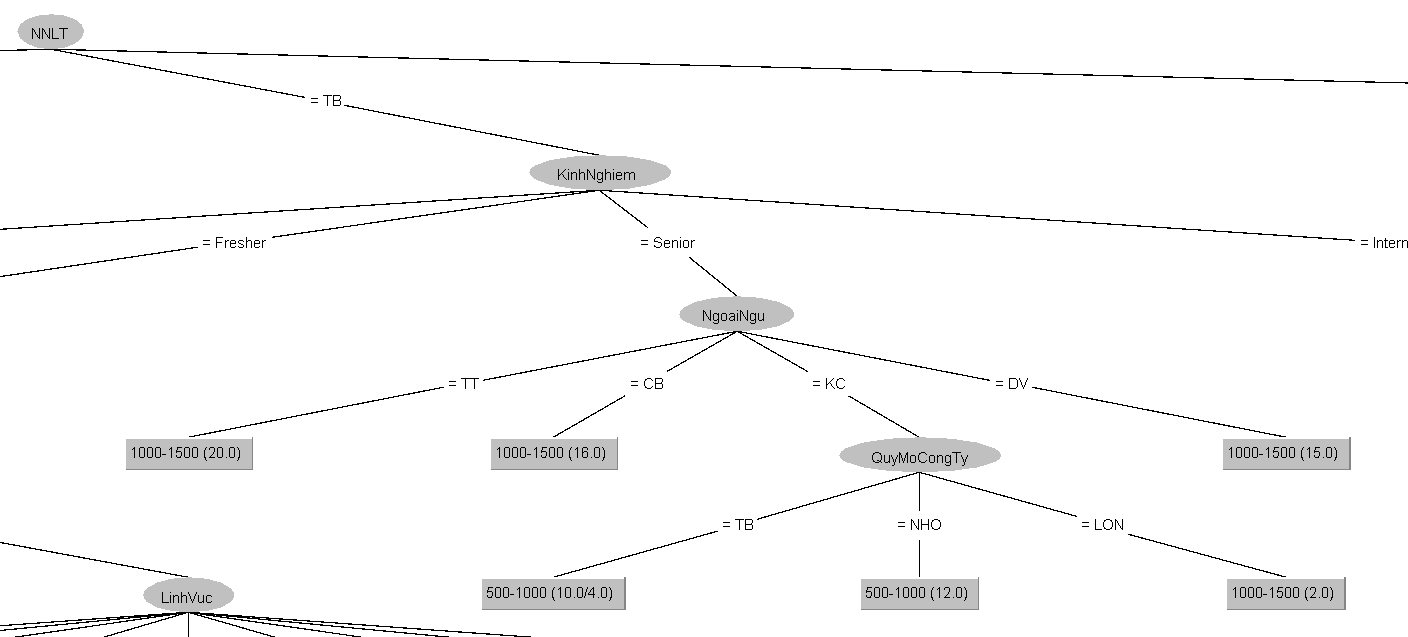
* Ngôn ngữ lập trình TB => Kinh nghiệm Fresher: pass



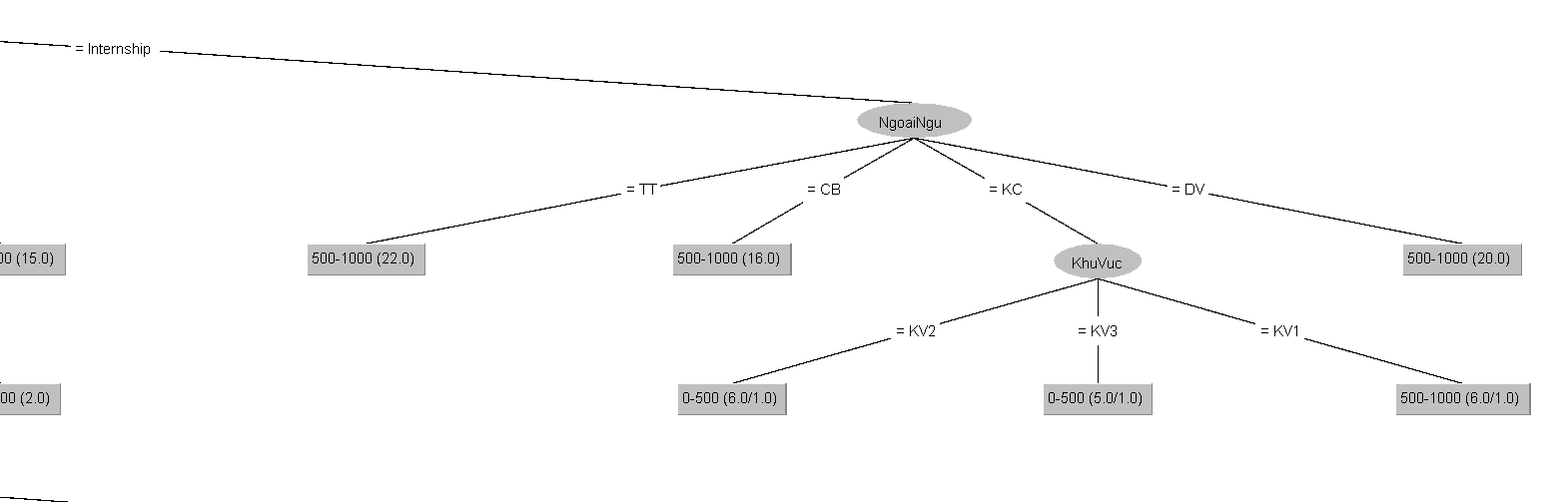
* Ngôn ngữ lập trình TB => Kinh nghiệm Junior => Khu vực KV3: pass
* Ngôn ngữ lập trình TB => Kinh nghiệm Junior => Khu vực KV2 => Quy mô công ty: pass



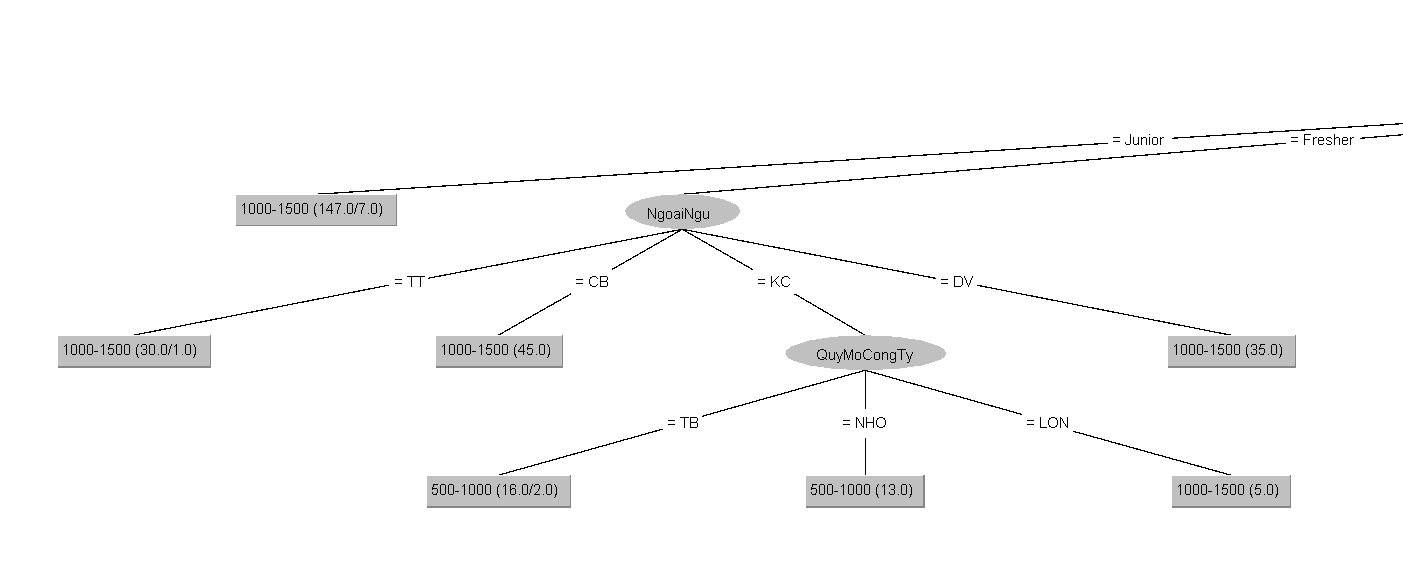
* Ngôn ngữ lập trình TB => Kinh nghiệm Junior => Khu vực KV3 => Ngoại ngữ: DV => Lĩnh vực : pass
* Ngôn ngữ lập trình TB => Kinh nghiệm Junior => Khu vực KV3 => Ngoại ngữ: KC, TT, CB: pass
* Ngôn ngữ lập trình TB => Kinh nghiệm Junior => Khu vực KV3 => Ngoại ngữ: DV => Lĩnh vực : pass



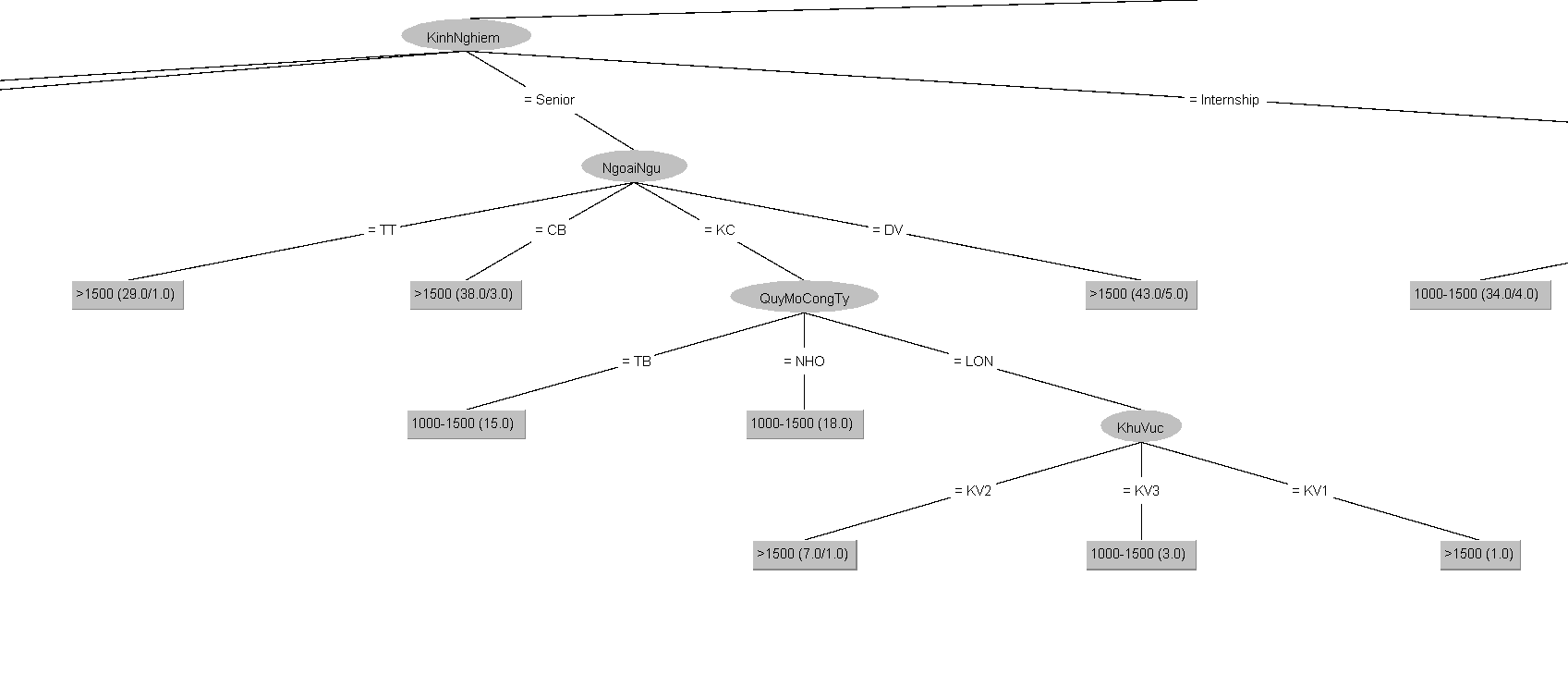
* Ngôn ngữ lập trình TB => Kinh nghiệm Senior => Ngoại ngữ KC => Quy mô công ty : pass
* Ngôn ngữ lập trình TB => Kinh nghiệm Senior => Ngoại ngữ DV, CB, TT: pass



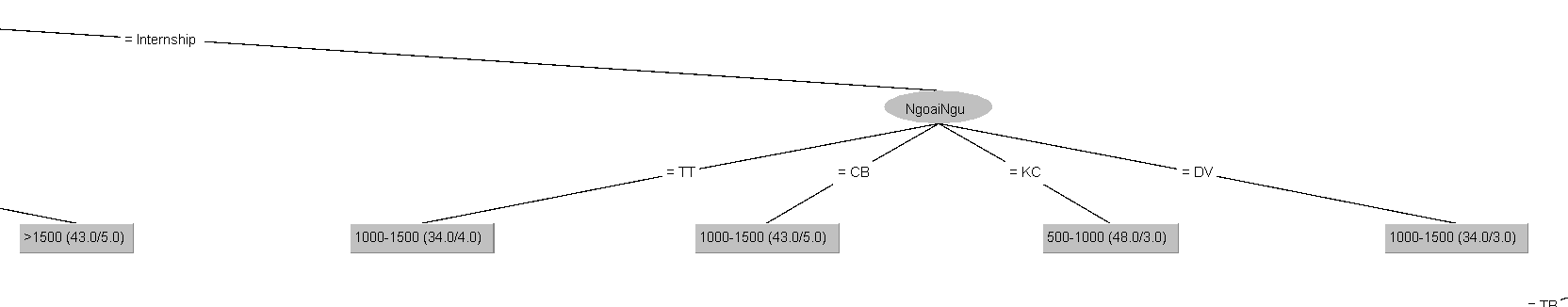
* Ngôn ngữ lập trình TB => Kinh nghiệm Intership=> Ngoại ngữ KC => Khu vực : pass
* Ngôn ngữ lập trình TB => Kinh nghiệm Intership => Ngoại ngữ DV, CB ,TT : pass



* Ngôn ngữ lập trình PB => Kinh nghiệm Junior: pass
* Ngôn ngữ lập trình PB => Kinh nghiệm Fresher => Ngoại ngữ KC => Quy mô công ty : pass
* Ngôn ngữ lập trình PB => Kinh nghiệm Fresher => Ngoại ngữ TT, CB, DV: pass



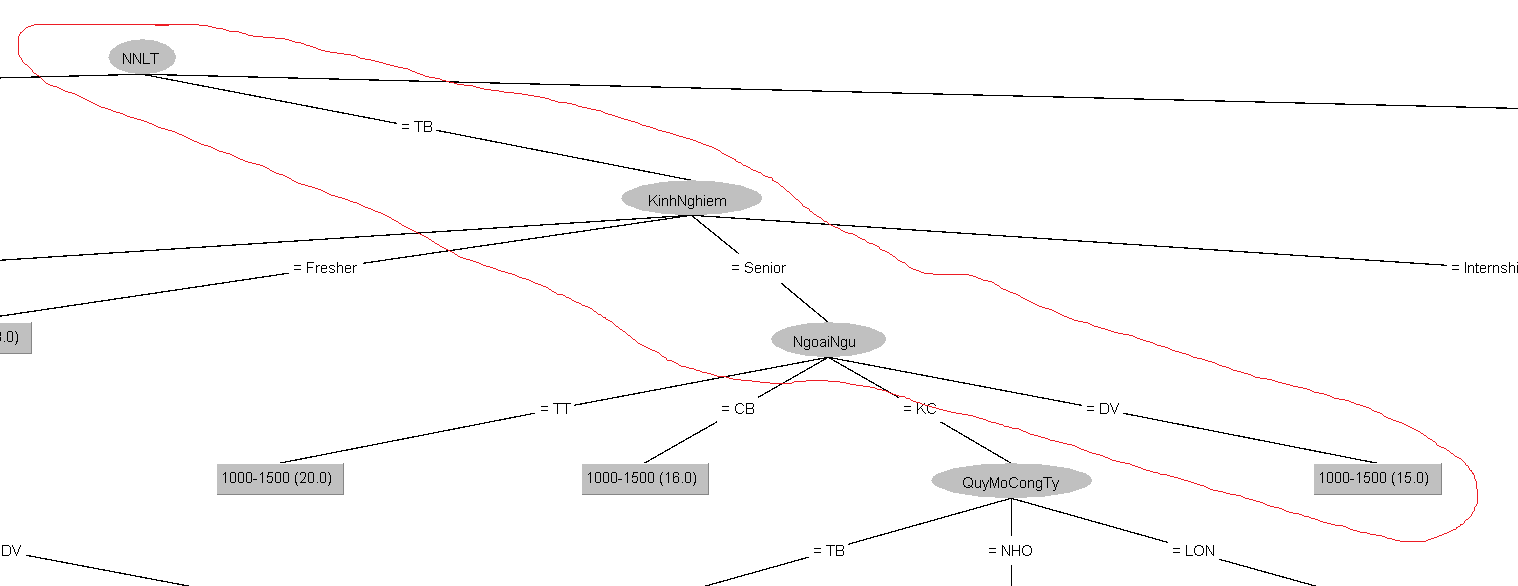
* Ngôn ngữ lập trình PB => Kinh nghiệm Senior => Ngoại ngữ DV, TT, CB : pass
* Ngôn ngữ lập trình PB => Kinh nghiệm Senior => Ngoại ngữ KC => Quy mô công ty : NHO, TB : pass
* Ngôn ngữ lập trình PB => Kinh nghiệm Senior => Ngoại ngữ KC => Quy mô công ty : LON => Khu vực: pass



* Ngôn ngữ lập trình PB => Kinh nghiệm Intership => Ngoại ngữ : pass

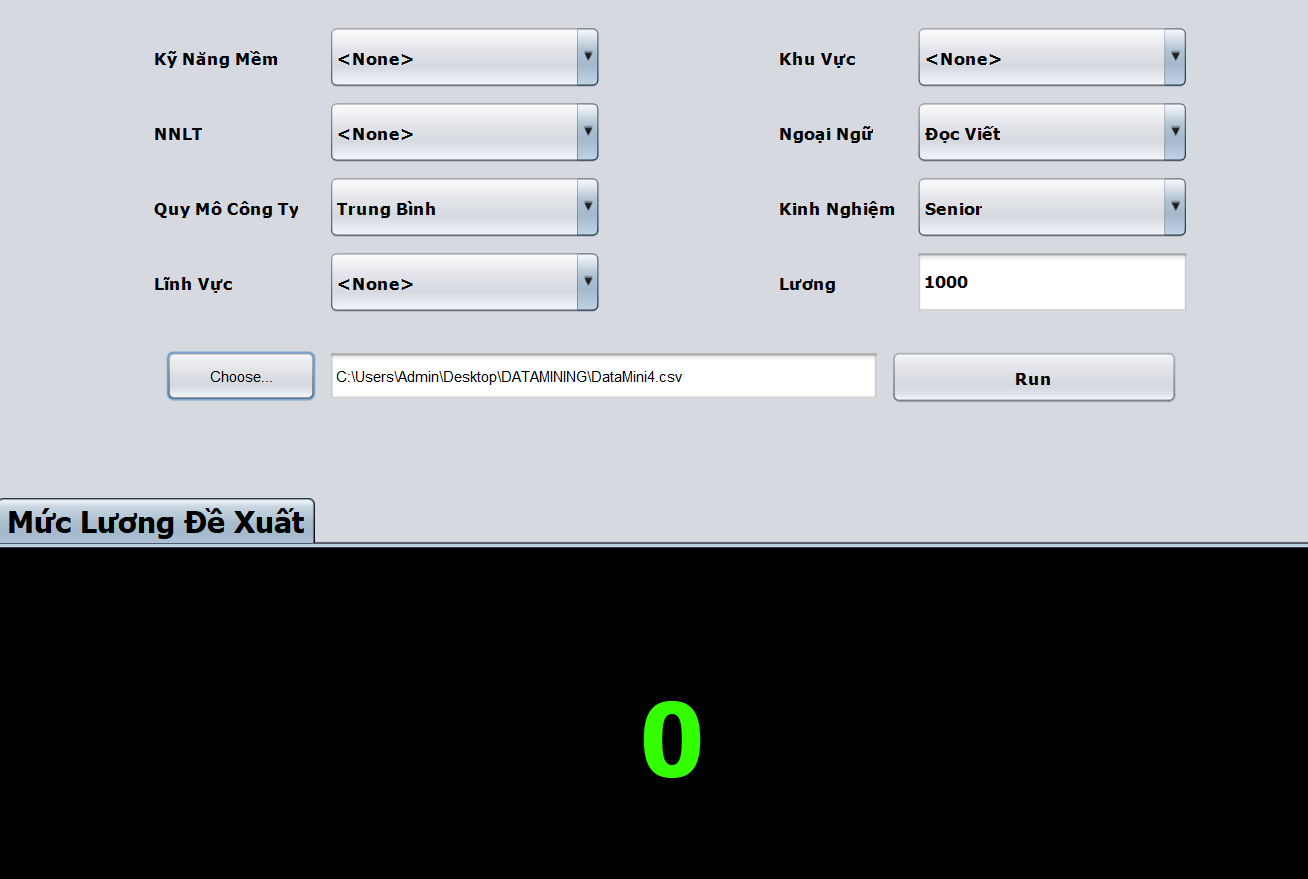
1. *Một số trường hợp chứng minh:*

* ***Trường hợp 1:***



* Chọn bộ dữ liệu gốc và nhập các thông tin như trên nhánh cây quyết định cần test và nhập lương mong muốn.

**(Trước khi chạy)**

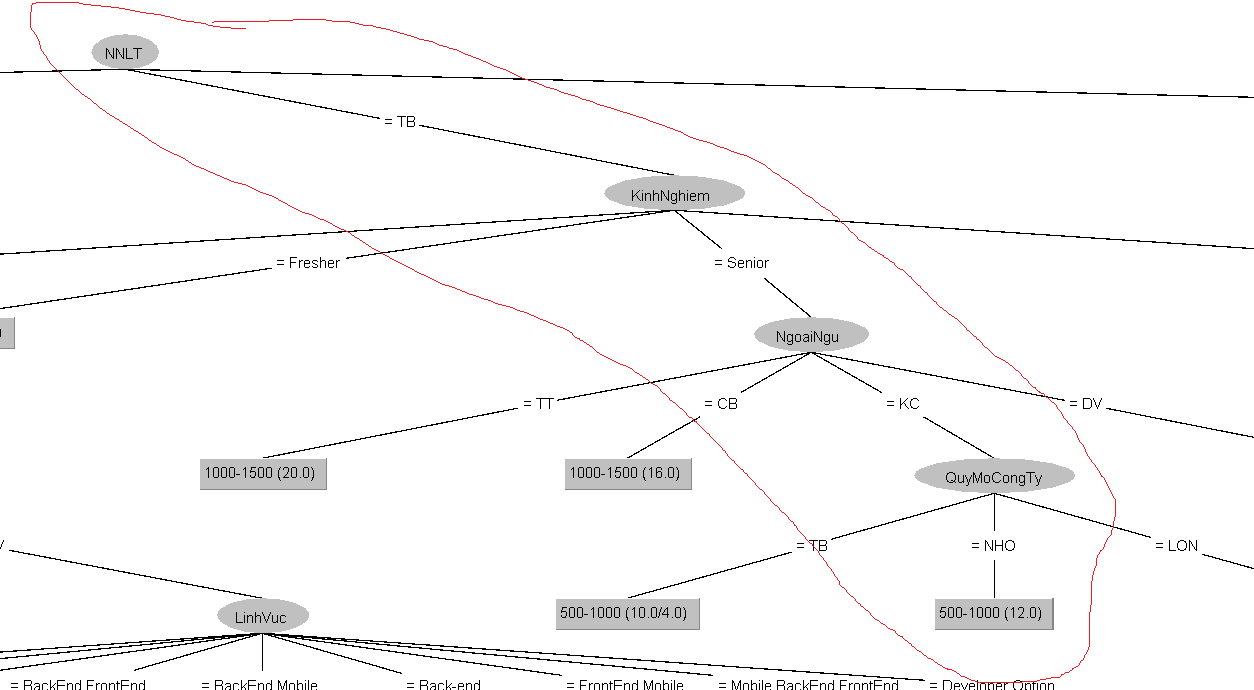


**(Sau khi chạy)**

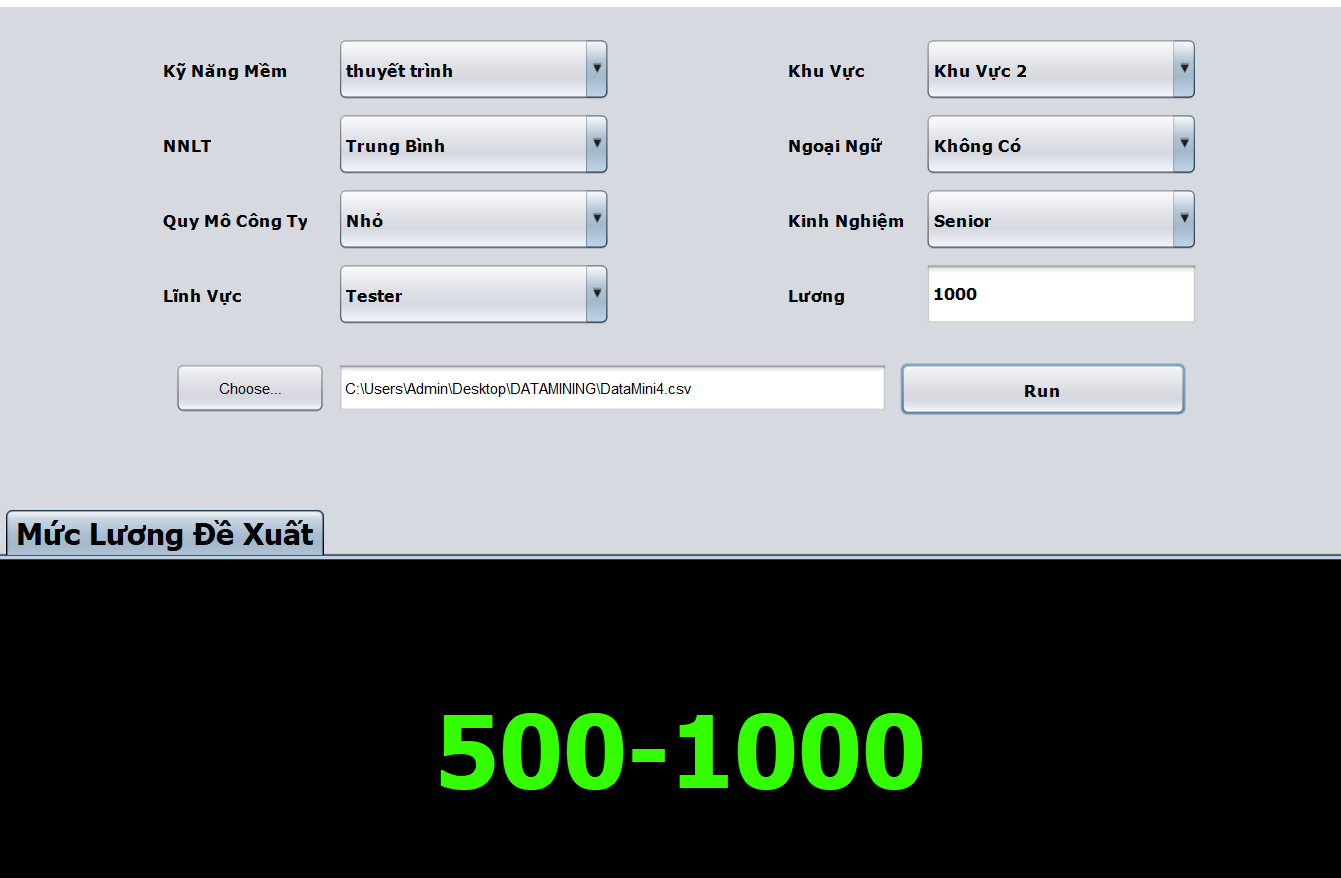


* *Nhận xét: Kết quả từ chương trình đúng như với trên cây quyết định. Một số trường bị khuyết sẽ được chương trình tự động điền.*
* ***Trường hợp 2:***

**(Nhánh cây cần kiểm tra)**

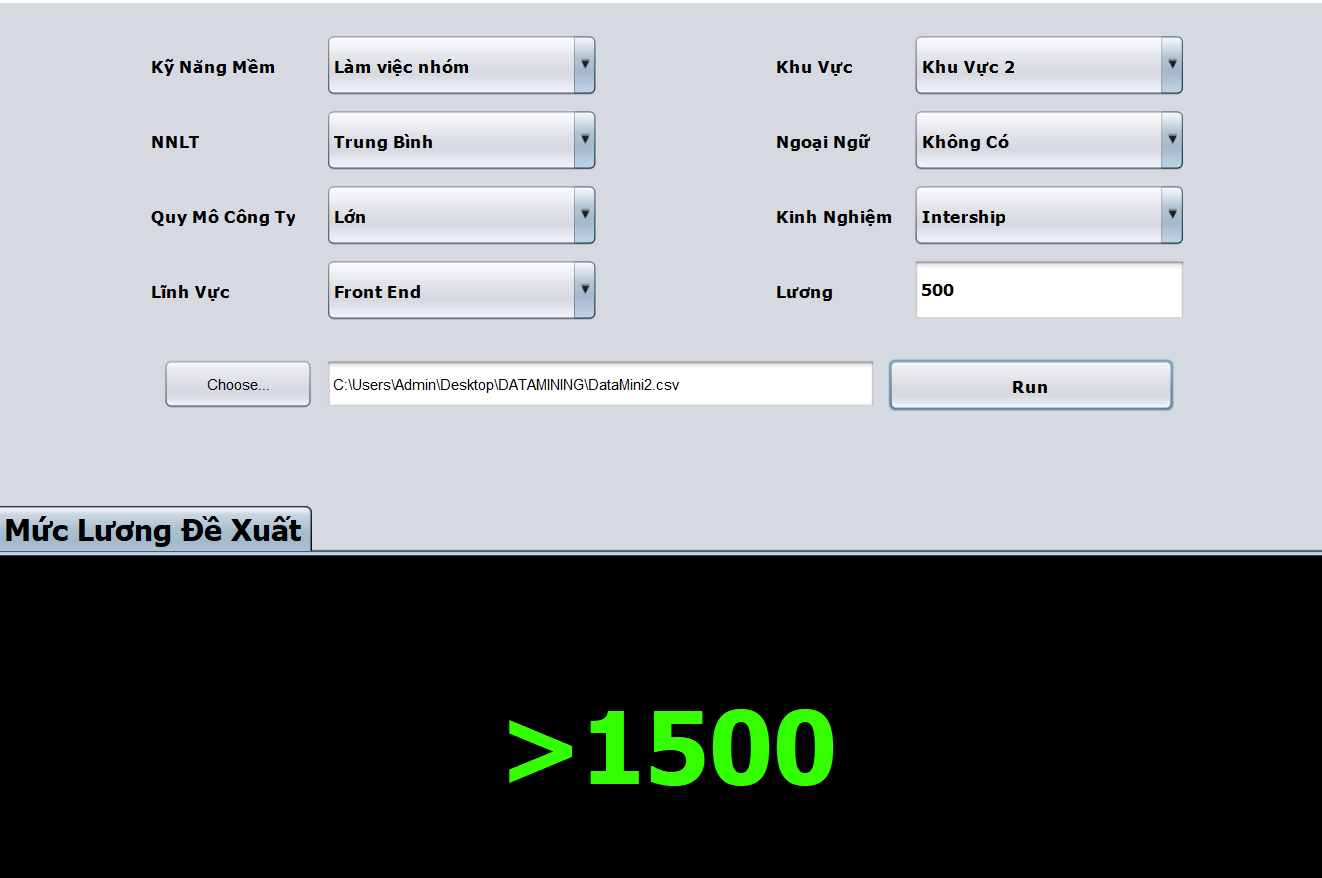


**(Kết quả kiểm tra)**



1. *Trên bộ dữ liệu thứ 2:*

* Kết quả test chương trình ra đúng như cây quyết định. Nhưng vì bộ dữ liệu thứ 2 có kết quả chính xác là 23,8% nên không đáng tin cậy.
* Ví dụ: Test trên nhánh cây : Kinh nghiệm Intership => Lĩnh vực Frontend => NNLT Trung bình => Lương >1500.

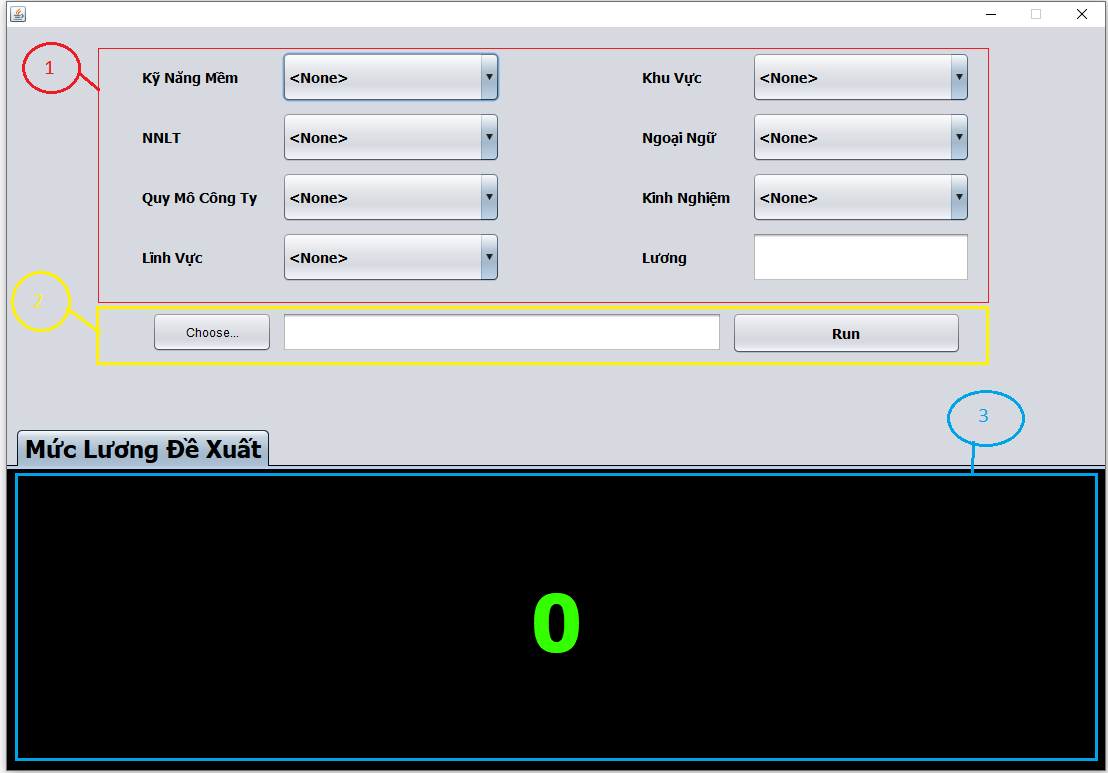


1. ***Xây dựng chương trình***

|  |  |  |
| --- | --- | --- |
|  | **Tên thư viện/công cụ** | **Mô tả** |
| Thư viện | **Weka.jar, Weka.src.jar** | Thư viện cung cấp các api thực thi các thuật toán thự thi huấn luyện dữ liệu và kiểm thử dữ liệu. |
| Công cụ xây dựng | **Netbean** | Công cụ **EDM** miễn phí, hổ trợ lập trình java, tích hợp thư viện và tạo giao diện sử dựng **java swing** cho ứng dụng. |

1. **Thư viện và cộng cụ xây dựng ứng dụng.**
2. **Thiết kế chương trình**

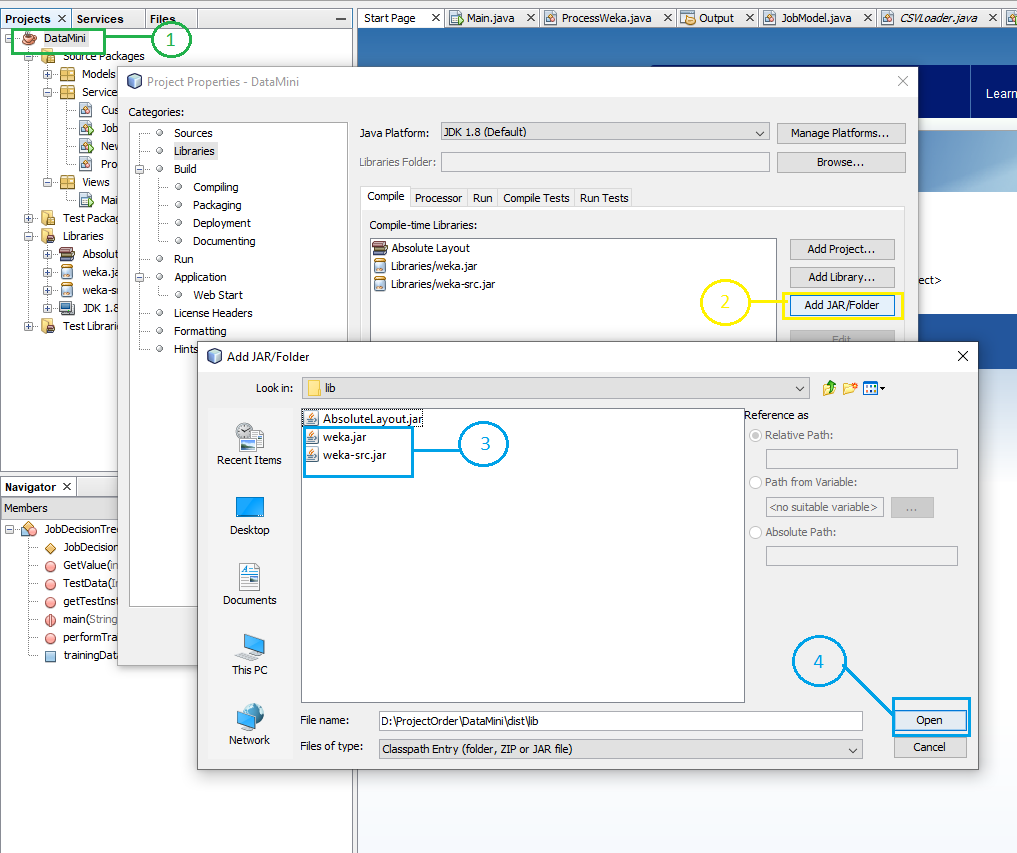
* Chương trình là 1 phần mềm hỗ trợ tư vấn lương cho người muốn tìm việc theo trình độ và nghề nghiệp
* Người dùng sẽ chọn những thuộc tính phù hợp với tình trạng bản thân và nhập vào mức lương mà mình mong muốn. Dựa trên những thuộc tính mà người dùng nhập vào chương trình sẽ đưa ra mức lương đề xuất để hỗ trợ cho người dùng trong việc tìm kiếm việc làm và offer mức lương tương ứng dựa trên trình độ và lĩnh vực của mình.

****** *Giao diện chương trình*

* Khu vực (1) : cho phép người dùng chọn các tùy chọn theo từng thuộc tính để thực hiện kiểm thử dữ liệu trên tập dữ liệu đã được huấn luyện.
* Khu vực (2) : chức năng cho phéo người dùng chọn tập dữ liệu muốn huấn luyện.
  + nút nhấn “Choose…” cho phéo mở một cửa sổ để chọn tập dữ liệu.
  + nút nhấn “Run” cho phép thực thi huấn luyện dữ liệu.
  + Chương trình sử dụng thuật toán **J48** được cung cấp vởi thư viện weka để thực hiện huấn luyện dữ liệu.
* Khu vực (3) : Kết quả sau khi thực hiện kiểm thử được hiện thị.

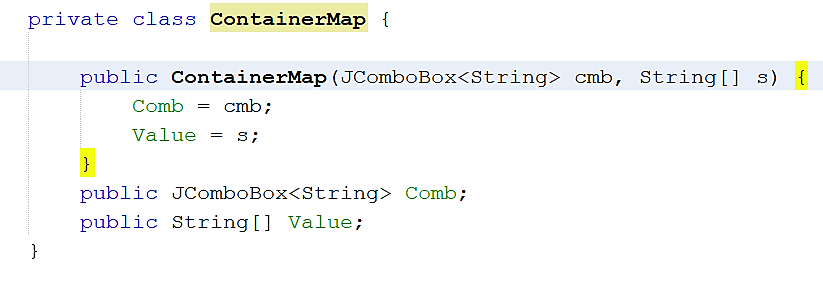
1. **Hiện thực (Tương tác với API) giải thuật và thử nghiệm**
2. *Tương tác với API*

*Connect API Weka*



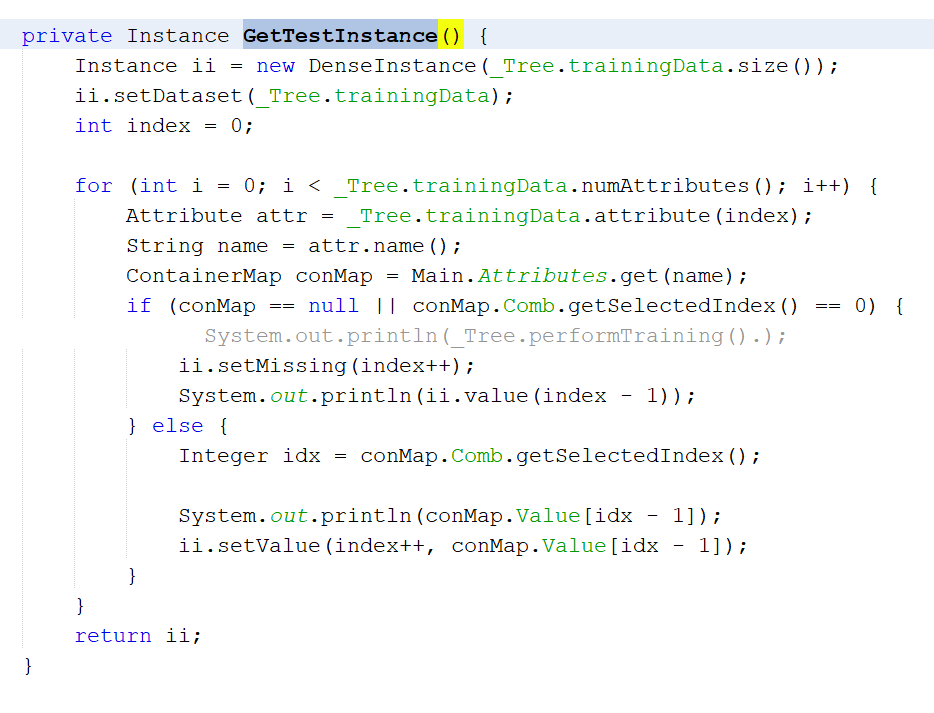
1. *Cấu trúc Project*

* Lớp **Main.java** : Thực hiện tạo các điều kiểm và sử lý các sự kiện trên frame.
* Lớp **ContainerMap** con nằm bên trong lớp **Main.java** : Thùng chứa các dữ liệu ánh xạ chuyển đổi dữ liệu về các thuộc tính phù hợp với các nhãn trong tập dữ liệu.

*class ContainerMap*

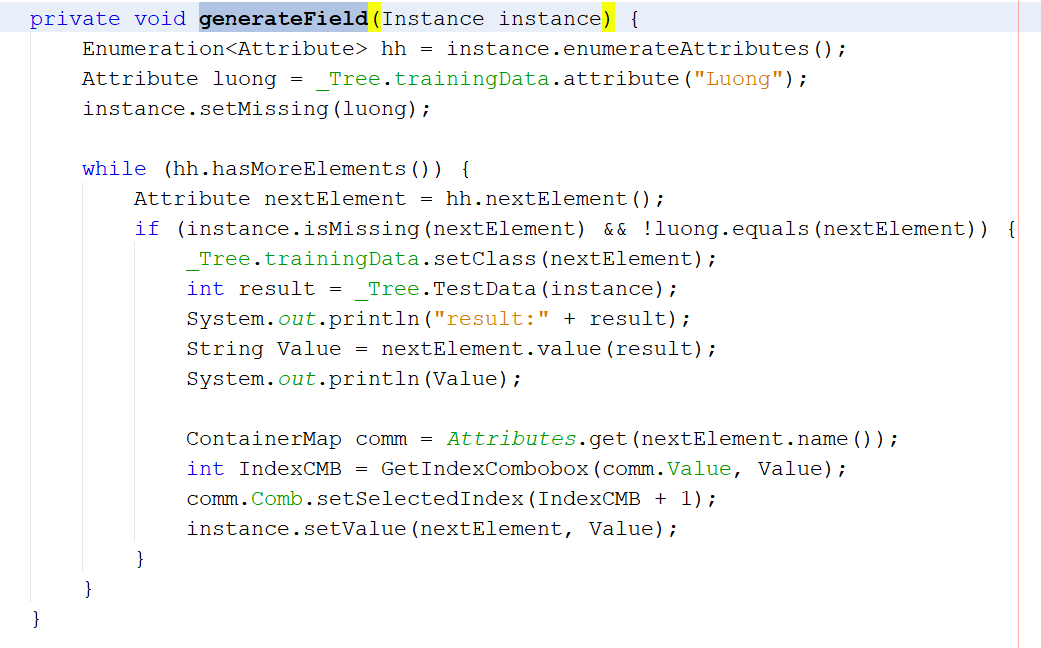
*Tạo thuộc tích Attributes và khởi tạo dữ liệu.*

* Thuộc tính **Attributes** chứa dữ liệu ánh xạ chuyển đổi và thực hiện khởi tạo trong phương thức contructor của **Lớp Main**: Trong đó key của **HashTable** se là tên của cột khi lấy từ tập dữ liệu đã được huấn luyện với thuật toán J48, Value của **HashTable** se là một thùng chứa **ContainerMap** chứa dữ liệu ánh xạ các thuộc tính của từ **Combobox** và các giá trị trong tập dự liệu.
* Phương thức **GetTestInstance** thực hiện chuyển đổi dữ liệu từ giao diện người dùng tương tác thàng kiểu một thực thể biểu thị dòng dữ liệu trong thư viện J48.

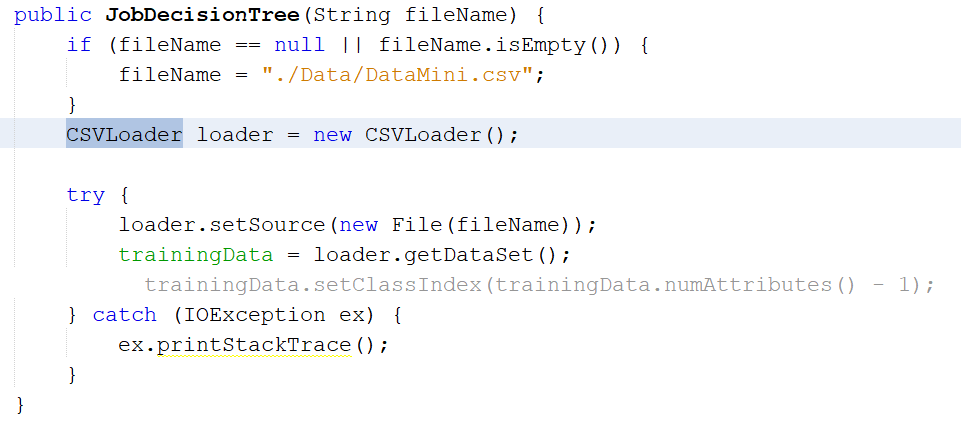
*Phương thức chuyển đổi dữ liệu từ giao diện thành thực thể cấu trúc trong J48*

* Phương thức **generateField** thực hiện tự động đề xuất các trường dữ liệu khuyết khi người dùng không điều đủ các trường dữ liệu.
* Chương trình sẽ thực hiện chạy huấn luyện trên tập dữ liệu đầu vào với cột mong muốn đề xuất là trường dữ liệu bị khuyết sau đó thực hiện kiểm thử dữ trên chính dòng dữ liệu do người dùng đưa vào kết quả đạt được sẽ là đề xuất đối với trường dữ liệu bị khuyết. thực hiện lần lượt trên tất cả các dòng dữ liệu bị khuyết ta sẽ được một dòng dữ liệu đầy đủ.

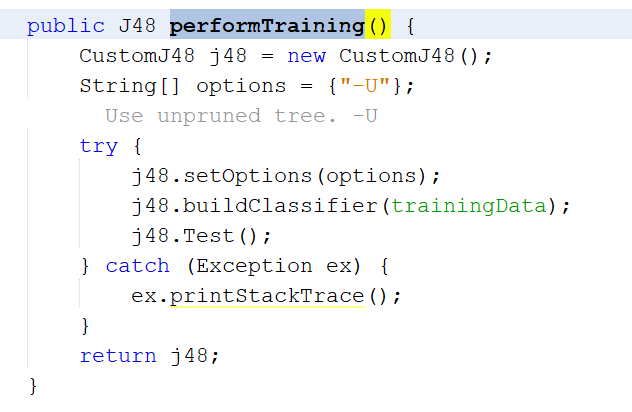
*Phương thức generateField*



* Lớp **JobDecisionTree.java** thực hiện các logic sử lý huấn luyện dư liệu và thực hiện các phướng thức thư viện trong thư viện **weka.rar**.
* Phướng thức constructor **JobDecisionTree** thực hiện đọc file sử dụng lớp **CSVLoader** của thư viện **Weka**.

*Phương thức constructor JobDecisionTree*.

* Phương thức **performTraining** thực hiện tạo ra một thưc thể thư viện thuật toán **J48**, thiết lập các tùy chình cho thuật toán và thực hiện train tập dữ liệu

*Phương thức performTraining.*

1. ***Kết luận***

* Hiện tại, nhóm chúng em về cơ bản đã xây dựng xong kho dữ liệu và hoàn thành ứng dụng. Chúng em xin cảm ơn sự hướng dẫn tận tình của thầy đã giúp chúng em hoàn thành đề tài.
* Tóm tắt quá trình và kết quả làm đề tài:
* **Về dữ liệu** chúng em nhận thấy vê kiên thức chuyên môn phân tích chỉ mới mức học thuật chưa có kinh nghiệm thực tiên , dữ liệu được lấy từ các trang web lơn như Top.dev,…. Nên về độ chính xác chúng em có thể tin tưởng được 50%.
* **Về test** tuy trong nhóm không có bạn nào chuyên về làm test nhưng chúng em đã có gắng học hỏi từ sự hướng dẫn của thầy và tự học hỏi để cải thiện ứng dụng và dữ liệu ít lỗi.
* **Về chương trình** chúng chỉ mới xây dựng chương trình với giao diện đơn giản.
* Do thời gian và kiến thức chuyên môn có hạn nên nhóm em biết rằng đề tài chắc chắn sẽ có nhiều sai sót, bên cạnh đó đây là một ứng dụng tri thức nên cũng cần thời gian dài để huấn luyện và hoàn chỉnh ứng dụng. Chúng em mong sẽ nhận được sự góp ý từ thầy sau khi chấm bài đê chúng em sẽ hoàn thiện hơn chương trình và bổ sung kinh nghiệm từ những thiếu sót đó.