GIẢI THUẬT RỪNG NGẪU NHIÊN VỚI LUẬT GÁN NHÃN CỤC BỘ CHO PHÂN LỚP

Đỗ Thanh Nghị, Phạm Nguyên Khang, Nguyễn Hữu Hòa, Nguyễn Minh Trung

Khoa CNTT-TT, Trường ĐHCT

dtnghi@cit.ctu.edu.vn

TÓM TẮT *- Trong bài viết này, chúng tôi đề xuất sử dụng luật gán nhãn cục bộ trong giải thuật rừng ngẫu nhiên để nâng cao hiệu quả phân lớp. Giải thuật rừng ngẫu nhiên của Breiman đề xuất là giải thuật phân lớp chính xác khi so sánh với các giải thuật học có giám sát hiện nay. Tuy nhiên, do sử dụng luật bình chọn số đông ở nút lá của cây quyết định làm dự báo của rừng ngẫu nhiên giảm hiệu quả. Để cải thiện kết quả dự báo của rừng ngẫu nhiên, chúng tôi đề xuất thay thế luật bình chọn số đông bởi luật gán nhãn cục bộ, k láng giềng. Kết quả thử nghiệm trên các tập dữ liệu gen từ website datam.i2r.a-star.edu.sg/datasets/krbd cho thấy rằng giải thuật rừng ngẫu nhiên sử dụng luật gán nhãn cục bộ do chúng tôi đề xuất cho kết quả phân loại tốt khi so sánh với rừng ngẫu nhiên của cây quyết định C4.5 và máy học véctơ hỗ trợ dựa trên các tiêu chí Precision, Recall, F1, Accuracy.*

Từ khóa *- Rừng ngẫu nhiên, cây quyết định, luật gán nhãn, luật cục bộ, k láng giềng, phân lớp dữ liệu nhiều chiều.*

I. GIỚI THIỆU

Phân lớp dữ liệu hay học có giám sát là một trong bốn nhóm bài toán quan trọng của khám phá tri thức và khai mỏ dữ liệu [Han et al., 2011]. Phân lớp dữ liệu xây dựng mô hình phân lớp từ tập dữ liệu có nhãn (lớp) đã được định nghĩa trước, để thực hiện gán nhãn tự động cho từng phần tử dữ liệu mới đến.

Phân lớp dữ liệu có số chiều lớn được biết là một trong 10 vấn đề khó của cộng đồng khai mỏ dữ liệu [Yang & Wu, 2006]. Mô hình học phân lớp thường cho kết quả tốt trong khi học nhưng lại cho kết quả rất thấp trong tập kiểm tra. Vấn đề khó khăn thường gặp chính là số chiều quá lớn và dữ liệu thường tách rời nhau trong không gian có số chiều lớn việc tìm mô hình phân lớp tốt có khả năng làm việc với dữ liệu có số chiều lớn là khó khăn do có quá nhiều khả năng lựa chọn mô hình. Việc tìm một mô hình phân lớp hiệu quả (phân lớp dữ liệu tốt trong tập thử) trong không gian giả thiết lớn là vấn đề khó. Đã có hai lớp giải thuật tiêu biểu, máy học véctơ hỗ trợ của Vapnik (SVM [Vapnik, 1995]) và rừng ngẫu nhiên của [Breiman, 2001], là những giải thuật phân lớp hiệu quả các tập dữ liệu có số chiều lớn.

Tiếp cận rừng ngẫu nhiên cho độ chính xác cao khi so sánh với các thuật toán học có giám sát hiện nay, bao gồm cả AdaBoost [Freund & Schapire, 1995], ArcX4 [Breiman, 1998] và SVM [Vapnik, 1995]. Khi xử lý dữ liệu cho có số chiều lớn, rừng ngẫu nhiên và SVM là hai giải thuật học nhanh, chịu đựng nhiễu tốt và không bị tình trạng học vẹt, điều này ngược lại với AdaBoost, ArcX4 rất dễ bị học vẹt và ảnh hưởng lớn với nhiễu [Grove & Schuurmans, 1998]. Tuy nhiên, luật quyết định ở nút lá của các cây trong rừng ngẫu nhiên dựa vào luật bình chọn số đông, điều này dẫn đến độ chính xác của giải thuật rừng ngẫu nhiên bị giảm khi phân lớp dữ liệu. Để khắc phục nhược điểm trên, chúng tôi đề xuất thay thế luật bình chọn số đông ở nút lá bằng luật gán nhãn cục bộ dựa trên giải thuật *k* láng giềng [Fix & Hodges, 1952]. Giải thuật rừng ngẫu nhiên sử dụng luật gán nhãn cục bộ do chúng tôi đề xuất thường cho kết quả phân lớp chính xác hơn so với giải thuật gốc. Kết quả thử nghiệm trên các tập dữ liệu gen [Jinyan & Huiqing, 2002] cho thấy rằng giải thuật rừng ngẫu nhiên cải tiến do chúng tôi đề xuất cho kết quả phân loại tốt khi so sánh với rừng ngẫu nhiên của cây quyết định C4.5 và máy học véctơ hỗ trợ dựa trên các tiêu chí *Precision*, *Recall*, *F1*, *Accuracy*.

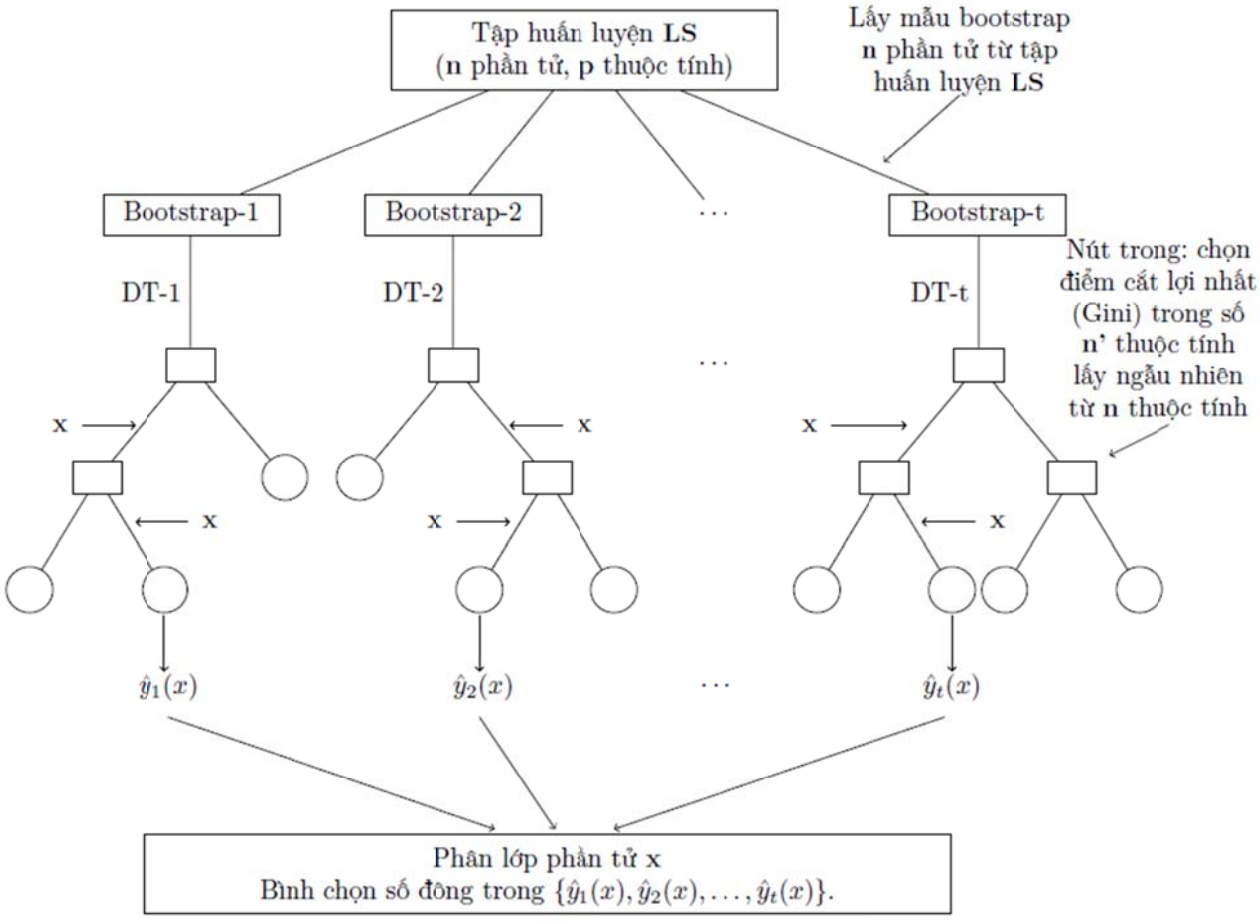
Phần còn lại của bài viết được tổ chức như sau. Chúng tôi sẽ trình bày tóm tắt giải thuật rừng ngẫu nhiên trong phần II, thay thế luật gán nhãn bình chọn số đông bằng luật gán nhãn cục bộ trong phần III. Kết quả thực nghiệm sẽ được trình bày trong phần IV. Phần thảo luận các nghiên cứu liên quan được trình bày trong phần V trước phần kết luận và hướng phát triển trong phần VI.

II. GIẢI THUẬT RỪNG NGẪU NHIÊN

Từ những năm 1990, cộng đồng máy học đã nghiên cứu cách để kết hợp nhiều mô hình phân loại thành tập hợp các mô hình phân loại để cho tính chính xác cao hơn so với chỉ một mô hình phân loại. Mục đích của các mô hình tập hợp là làm giảm thành phần lỗi variance và/hoặc bias của các giải thuật học. Bias là khái niệm về lỗi của mô hình học (không liên quan đến dữ liệu học) và variance là lỗi do tính biến thiên của mô hình so với tính ngẫu nhiên của các mẫu dữ liệu học. [Buntine, 1992] đã giới thiệu các kỹ thuật Bayes để giảm variance của các phương pháp học. Phương pháp xếp chồng [Wolpert, 1992] hướng tới việc cực tiểu hóa bias của các giải thuật học. Trong khi [Freund & Schapire, 1995] đưa ra Boosting, [Breiman, 1998] đề nghị ArcX4 để cùng giảm bias và variance, còn Bagging [Breiman, 1996] thì giảm variance của giải thuật học nhưng không làm tăng bias quá nhiều. Tiếp cận rừng ngẫu nhiên [Breiman, 2001] là một trong những phương pháp tập hợp mô hình thành công nhất. Giải thuật rừng ngẫu nhiên xây dựng cây không cắt nhánh nhằm giữ cho bias thấp và dùng tính ngẫu nhiên để điều khiển tính tương quan thấp giữa các cây trong rừng.

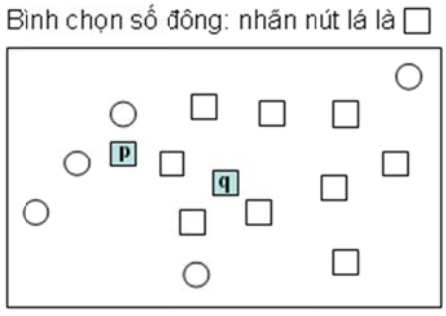
Rừng ngẫu nhiên (được mô tả trong hình 1) tạo ra một tập hợp các cây quyết định không cắt nhánh, mỗi cây được xây dựng trên tập mẫu bootstrap (lấy mẫu ngẫu nhiên có hoàn lại), tại mỗi nút phân hoạch tốt nhất được thực hiện từ việc chọn ngẫu nhiên một tập con các thuộc tính.

Lỗi tổng quát của rừng phụ thuộc vào độ chính xác của từng cây thành viên trong rừng và sự phụ thuộc lẫn nhau giữa các cây thành viên. Giải thuật rừng ngẫu nhiên cho độ chính xác cao khi so sánh với các thuật toán học có giám sát hiện nay, chịu đựng nhiễu tốt.



Hình 1. Giải thuật rừng ngẫu nhiên cho phân lớp dữ liệu

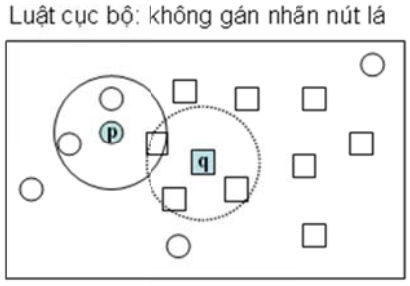
Tuy nhiên, nếu chúng ta trở lại luật gán nhãn ở nút lá của các cây quyết định trong rừng ngẫu nhiên, 2 giải thuật cây quyết định phổ biến là CART [Breiman et al., 1984] và C4.5 [Quinlan, 1993] thường dùng chiến lược bình chọn số đông. Thời điểm xây dựng xây dựng cây quyết định, nếu nút lá có chứa các phần tử dữ liệu của các lớp không thuần nhất, việc gán nhãn cho nút lá được tính cho nhãn của lớp có số lượng phần tử lớn nhất chứa trong nút lá. Xét ví dụ như hình 2, nút lá có chứa 14 phần tử trong đó lớp hình vuông có 9 phần tử và lớp hình tròn có 5 phần tử. Nút lá sẽ được gán nhãn là hình vuông do số phần tử lớp hình vuông nhiều hơn hình tròn. Chiến lược gán nhãn này làm cho luật quyết định không được chính xác. Khi phân lớp, phần tử nào rơi vào nút lá đều được gán nhãn của nút lá. Vì vậy, phần tử ***p***, ***q*** được gán nhãn là vuông. Hiệu quả phân lớp không cao (dự đoán nhãn của phần tử ***p*** có thể sai).



Hình 2. Luật gán nhãn bình chọn số đông ở nút lá của cây quyết định (nút lá có nhãn là vuông), điểm p và q được phân lớp vuông

III. LUẬT GÁN NHÃN CỤC BỘ

Để nâng cao hiệu quả phân lớp của cây quyết định trong giải thuật rừng ngẫu nhiên, chúng tôi đề xuất thay thế luật gán nhãn trên cơ sở bình chọn số đông bởi luật gán nhãn cục bộ với giải thuật *k* láng giềng [Fix & Hodges, 1952]. Thay vì việc gán nhãn ở nút lá được thực hiện khi xây dựng cây, chúng tôi trì hoãn việc gán nhãn này. Nghĩa là nút lá vẫn chưa được gán nhãn. Chúng tôi chỉ thực hiện việc gán nhãn trong khi dự báo phần tử mới đến. Xét tại nút lá như hình 3 vẫn chưa được gán nhãn. Với luật quyết định cục bộ dựa trên *3* láng giềng. Khi phần tử dữ liệu mới đến chẳng hạn như ***p*** và ***q***, rơi vào cùng nút lá; chúng tôi thực hiện tìm *3* phần tử trong nút lá gần nhất với dữ liệu mới đến, sau đó mới thực hiện việc gán nhãn cho phần tử cần dự báo được dựa trên nhãn của các láng giềng. Khi phân lớp, phần tử ***p*** rơi vào nút lá, chúng ta tìm *3* láng giềng của ***p***, gán nhãn cho ***p*** dựa trên bình chọn số đông từ *3* láng giềng, nhãn của ***p*** được gán là tròn. Tương tự, phần tử ***q*** được gán nhãn là vuông từ bình chọn số đông từ *3* láng giềng của nó. Luật quyết định này giúp cho việc phân lớp của cây đạt chính xác cao hơn vì trong chiến lược này, mặc dù các phần tử dự báo rơi vào cùng nút lá nhưng nhãn của nó có thể khác nhau trong khi chiến lược bình chọn số đông thường sử dụng trong cây quyết định lại gán cùng nhãn cho các phần tử rơi vào cùng nút lá.

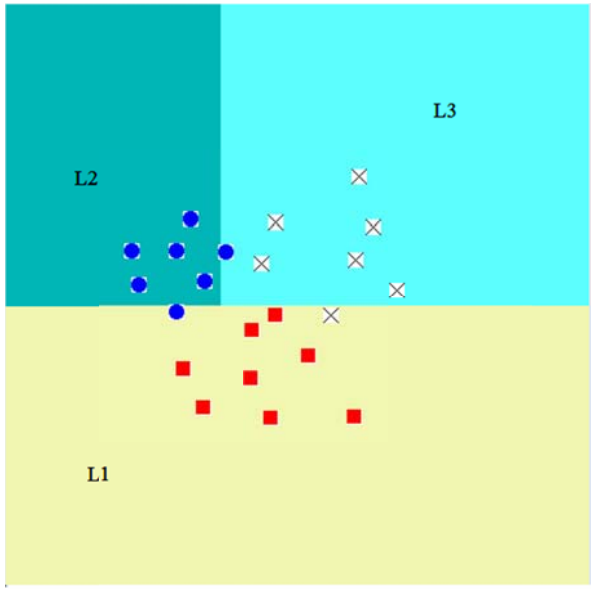


Hình 3. Luật gán nhãn cục bộ (3 láng giềng) ở nút lá của cây quyết định (nút lá chưa được gán nhãn), điểm p, q được gán nhãn lần lượt là tròn, vuông dựa trên bình chọn số đông của 3 láng giềng

Để minh họa sự ảnh hưởng đến mô hình phân lớp của cây quyết định khi thay thế luật gán nhãn ở nút lá. Chúng ta có thể xét ví dụ phân lớp (3 lớp gồm *tròn*, *vuông*, *chéo*) như trong hình 4.

Giải thuật học cây quyết định sử dụng luật bình chọn số đông để gán nhãn ở nút lá, C4.5 [Quinlan, 1993] huấn luyện mô hình phân lớp trên tập dữ liệu này, sinh ra mô hình có biên giới tách lớp là các hình chữ nhật như hình 4. Các phần tử rơi vào vùng *L1*, được gán nhãn là *vuông*. Các phần tử trong vùng *L2* được gán nhãn là *tròn*, trong khi các phần tử ở phân vùng *L3* được gán nhãn là *chéo*.

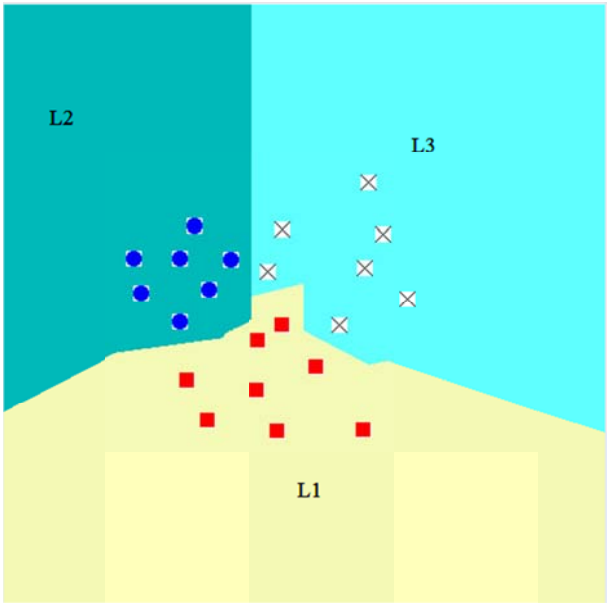
Có thể thấy rằng lớp *tròn* có 2 phần tử bị gán nhãn sai, 1 phần tử bị gán nhãn sang lớp *vuông* và 1 bị gán nhãn lớp *chéo*. Lớp *chéo* cũng có 1 phần tử bị gán nhãn sai sang lớp *vuông*.



Hình 4. Cây quyết định sử dụng luật bình chọn số đông để gán nhãn ở nút lá

Giải thuật học cây quyết định sử dụng luật gán nhãn cục bộ *1* láng giềng ở nút lá, huấn luyện mô hình phân lớp trên tập dữ liệu này, sinh ra mô hình có biên giới tách lớp mềm dẽo (có thể không là hình chữ nhật) như hình 5. Các phần tử rơi vào vùng *L1*, được gán nhãn là *vuông*. Các phần tử trong vùng *L2* được gán nhãn là *tròn*, trong khi các phần tử ở phân vùng *L3* được gán nhãn là *chéo*.

Có thể thấy rằng các phần tử của tập dữ liệu đều được mô hình gán nhãn chính xác với lớp của nó. Điều này chứng tỏ việc thay thế luật gán nhãn ở nút lá giúp cho mô hình cây quyết định được sử dụng trong giải thuật rừng ngẫu nhiên trở nên hiệu quả hơn.



Hình 5. Cây quyết định sử dụng luật cục bộ (1 láng giềng) để gán nhãn ở nút lá

IV. KẾT QUẢ THỰC NGHIỆM

Để có thể đánh giá hiệu quả của giải thuật rừng ngẫu nhiên sử dụng luật gán nhãn cục bộ, chúng tôi cài đặt giải thuật rừng ngẫu nhiên cây quyết định C4.5 sử dụng luật bình chọn số đông để gán nhãn ở nút lá, RF-C4.5(Maj) và giải thuật sử dụng luật gán nhãn cục bộ *k* láng giềng ở nút lá, RF-C4.5(*k*NN), bằng ngôn ngữ lập trình C/C++ có kế thừa từ mã nguồn của C4.5 được cung cấp bởi [Quinlan, 1993].

Bảng 1. Mô tả các tập dữ liệu gen

ID	Tập dữ liệu	Số phần tử	Số chiều	Lớp	Nghi thức
1	ALL-AML-Leukemia	72	7129	ALL, AML	trn-tst
2	MLL-Leukemia	72	12582	MLL, rest	trn-tst
3	Breast Cancer	97	24481	relapse, non-relapse	trn-tst
4	Prostate Cancer	136	12600	cancer, normal	trn-tst
5	Lung Cancer	181	12533	cancer, normal	trn-tst
6	Diffuse Large B-Cell Lymphoma	47	4026	germinal, activated	loo
7	Subtypes of Acute Lymphoblastic (Hyperdip)	327	12558	Hyperdip, rest	trn-tst
8	Subtypes of Acute Lymphoblastic (TEL-AML1)	327	12558	TEL-AML1, rest	trn-tst
9	Subtypes of Acute Lymphoblastic (T-ALL)	327	12558	TEL-ALL, rest	trn-tst
10	Subtypes of Acute Lymphoblastic (Others)	327	12558	Others, diagnostic groups	trn-tst

Dữ liệu dùng trong thực nghiệm là 10 tập dữ liệu gen có số chiều rất lớn, được lấy tại [Jinyan & Huiqing, 2002]. Bên cạnh đó, chúng tôi quan sát kết quả của giải thuật chúng tôi đề xuất RF-C4.5(*k*NN) trong thực nghiệm bằng cách so sánh với giải thuật RF-C4.5(Maj) và máy học véctơ hỗ trợ LibSVM [Chang & Lin, 2011]. Tất cả các kết quả đều được thực hiện trên một máy tính cá nhân (Intel Core2 Duo 2.4 GHz, 4GB RAM) chạy hệ điều hành Linux Mandriva.

Chúng tôi tiến hành thực nghiệm trên 10 tập dữ liệu gen có số chiều rất lớn từ kho dữ liệu sinh-y học. Mô tả các tập dữ liệu được tìm thấy trong bảng 1. Chúng tôi chú ý đến các nghi thức kiểm tra được liệt kê trong cột cuối của bảng 1. Với những tập dữ liệu có sẵn tập học và tập kiểm tra, chúng tôi dùng tập học để thử điều chỉnh các tham số ở đầu vào của các giải thuật nhằm thu được độ chính xác tốt khi học. Sau đó, dùng mô hình thu được để phân lớp tập kiểm tra. Nếu tập học và tập kiểm tra không có sẵn, các nghi thức kiểm tra chéo (cross-validation protocol) để đánh giá. Do các tập dữ liệu có ít hơn 300 phần tử, chúng tôi dùng nghi thức kiểm tra chéo leave-one-out (loo). Tức là dùng một phần tử trong tập dữ liệu để làm tập kiểm tra, các phần tử khác dùng để học. Lặp lại đến khi tất cả các phần tử đều được dùng để kiểm thử một lần.

Để thấy rõ hơn tính hiệu quả của RF-C4.5(*k*NN) so với RF-C4.5(Maj) và LibSVM, chúng tôi tiến hành so sánh hiệu quả của các thuật toán phân lớp dựa trên các tiêu chí như *Precision*, *Recall*, *F1-measure* và *Accuracy* [van Rijsbergen, 1979].

- *Precision* của một lớp là số phần tử dữ liệu được phân lớp đúng về lớp này chia cho tổng số phần tử dữ liệu được phân về lớp này.
- *Recall* của một lớp là số phần tử dữ liệu được phân lớp đúng về lớp này chia cho tổng số phần tử dữ liệu của lớp.
- *F1-measure* là tổng hợp của *Precision* và *Recall* và được định nghĩa là hàm trung bình điều hòa giữa hai giá trị *Precision* và *Recall*:

$$F1 = \frac{2 \times \mathrm{Pr}ecision \times \mathrm{Re}call}{\mathrm{Pr}ecision + \mathrm{Re}call}$$

- Độ chính xác *Accuracy* là số điểm dữ liệu được phân lớp đúng của tất cả các lớp chia cho tổng số điểm dữ liệu.

Khi xây dựng mô hình, các giải thuật rừng ngẫu nhiên xây dựng *200* cây quyết định cho tất cả các tập dữ liệu. Luật gán nhãn cục bộ sử dụng *1* láng giềng. Riêng máy học LibSVM chỉ cần sử dụng hàm nhân tuyến tính là phân lớp tốt nhất các tập dữ liệu gen. Chúng tôi thu được kết quả của các giải thuật như trình bày trong bảng 2 (*Precision*, *Recall*, *F1*), bảng 3 (*Accuracy*).

Bảng 2. Kết quả phân lớp của LibSVM, RF-C4.5(Maj) và RF-C4.5(*k*NN)

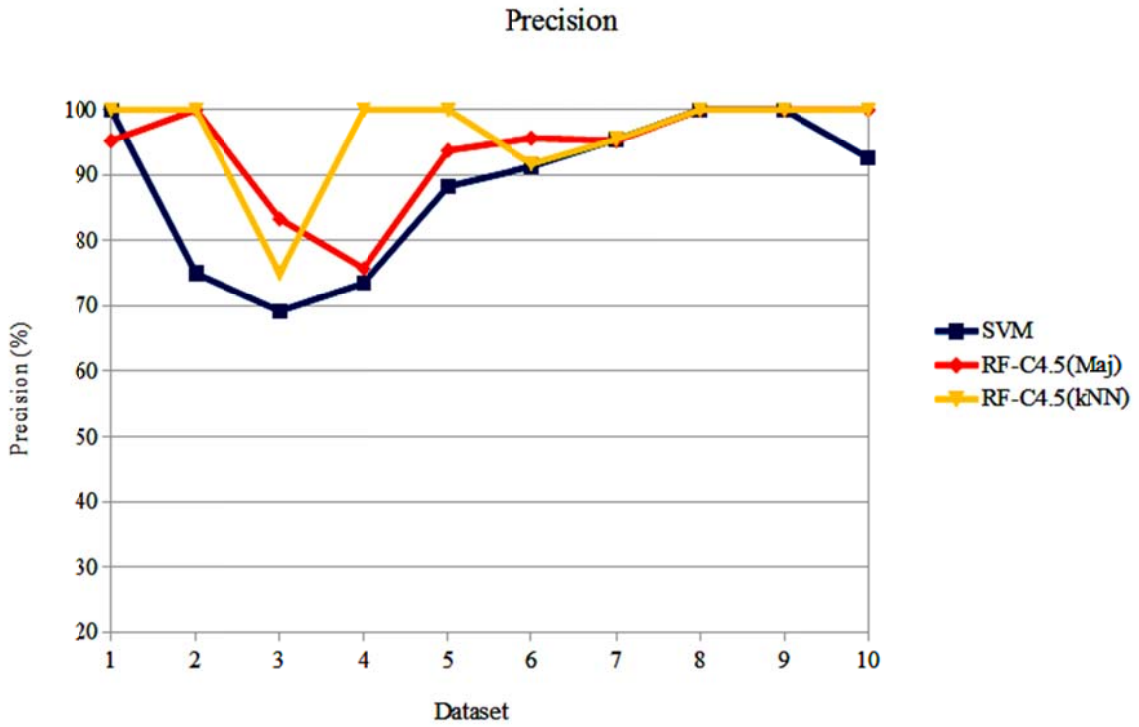
ID	Precision			Recall			F1-measure		
	Lib-SVM	**RF-C4.5 (Maj)**	**RF-C4.5 (*k*NN)**	**Lib-SVM**	**RF-C4.5 (Maj)**	**RF-C4.5 (*k*NN)**	**Lib-SVM**	**RF-C4.5 (Maj)**	**RF-C4.5 (*k*NN)**
1	**100**	95.24	**100**	95	**100**	95	97.44	**97.56**	97.44
2	75	**100**	**100**	**100**	**100**	**100**	85.71	**100**	**100**
3	69.23	**83.33**	75	75	83.33	**85.71**	72	**83.33**	80
4	73.53	75.76	**100**	**100**	**100**	66.67	84.75	**86.21**	75
5	88.26	93.75	**100**	**100**	**100**	**100**	93.75	96.77	**100**
6	91.3	**95.65**	91.67	87.5	91.67	**95.65**	89.36	**93.62**	**93.62**
7	**95.45**	95.24	**95.45**	**95.45**	90.91	**95.45**	**95.45**	93.02	**95.45**
8	**100**	**100**	**100**	**100**	96.3	**100**	**100**	98.11	**100**
9	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
10	92.59	**100**	**100**	39.68	29.63	**74.07**	55.56	45.71	**76.92**

Nhìn vào bảng 2, 3 và các đồ thị ở hình 6, 7, 8, 9, về kết quả phân lớp để so sánh hiệu quả của giải thuật LibSVM, RF-C4.5(Maj) và RF-C4.5(*k*NN).

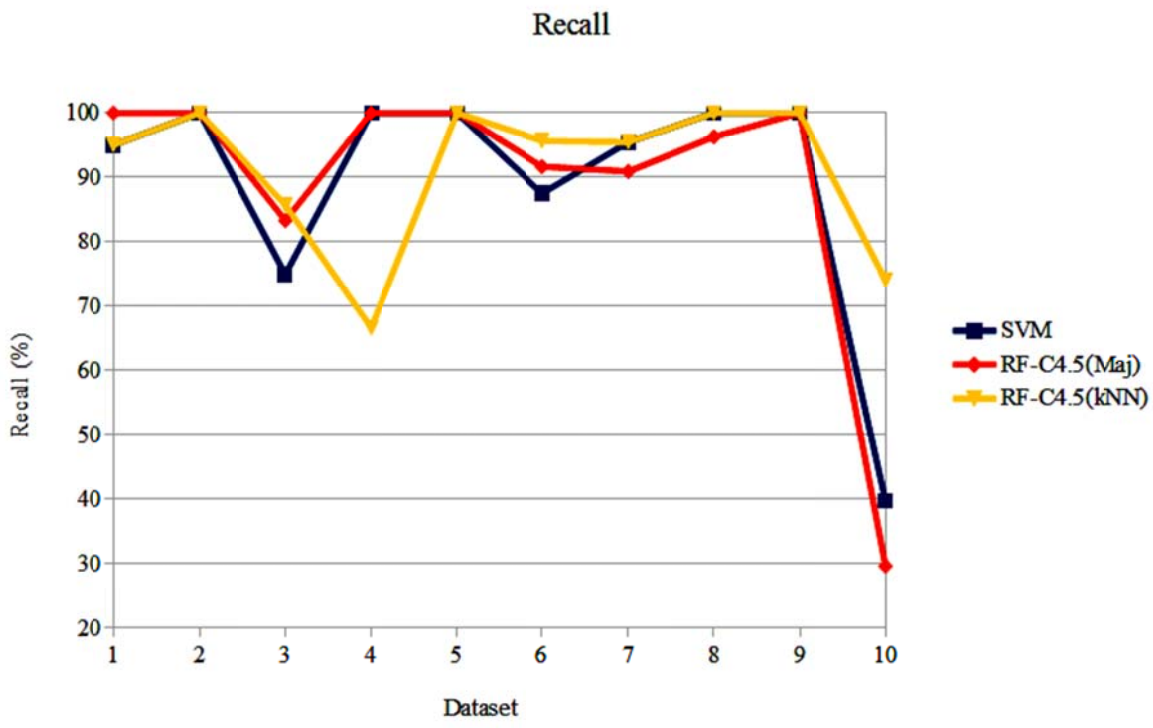
Chúng ta có thể thấy rằng với tiêu chí *Precision*, giải thuật RF-C4.5(*k*NN) cho kết quả tốt nhất *8/10* tập dữ liệu. Khi so sánh dựa vào tiêu chí *Recall*, RF-C4.5(*k*NN) cũng cho kết quả tốt *8/10* tập dữ liệu.

Xét trên tiêu chí *F1* (trung bình điều hòa giữa hai giá trị *Precision* và *Recall*), RF-C4.5(*k*NN) cho kết quả tốt nhất *7/10* tập dữ liệu khi so sánh với LibSVM và RF-C4.5(Maj).

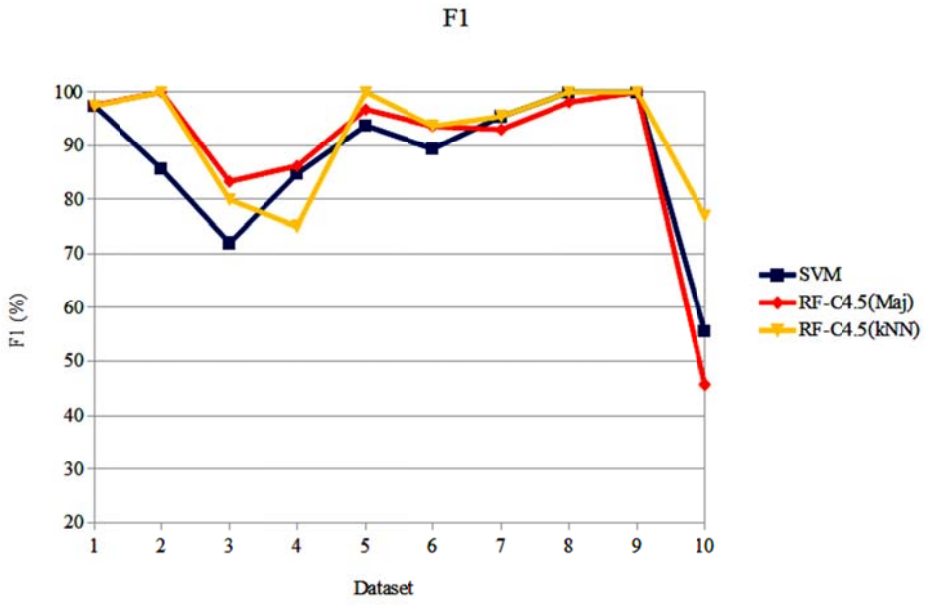
Giải thuật RF-C4.5(*k*NN) có độ chính xác toàn cục (*Accuracy*) cao nhất trên tất cả 10 tập dữ liệu khi so sánh với LibSVM và RF-C4.5(Maj).



Hình 6. So sánh tiêu chí Precision của 3 giải thuật trên 10 tập dữ liệu



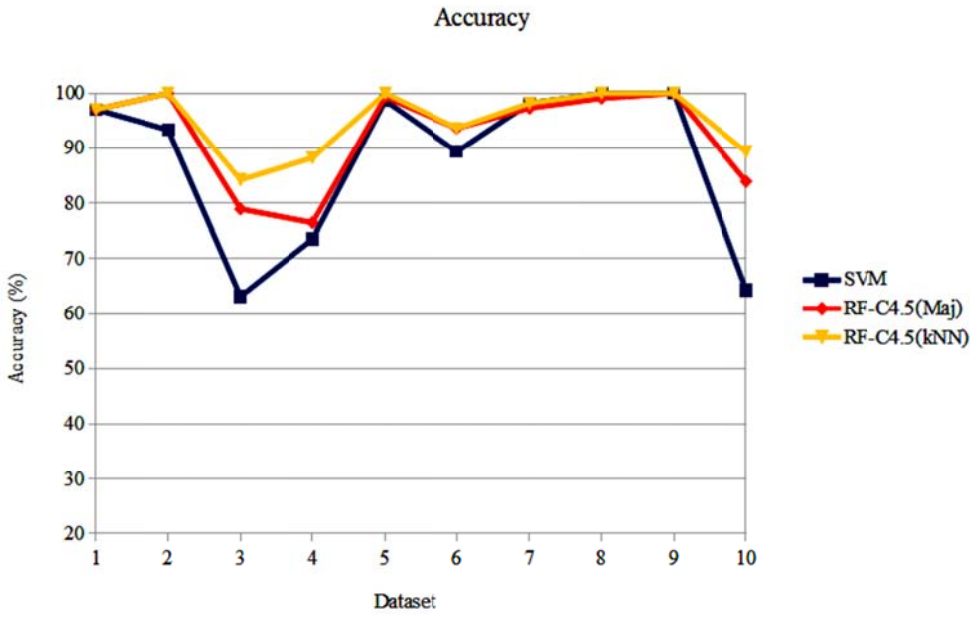
Hình 7. So sánh tiêu chí Recall của 3 giải thuật trên 10 tập dữ liệu



Hình 8. So sánh tiêu chí F1 của 3 giải thuật trên 10 tập dữ liệu

Bảng 3. Kết quả phân lớp của LibSVM, RF-C4.5(Maj) và RF-C4.5(*k*NN)

ID	Accuracy		
	Lib-SVM	**RF-C4.5(Maj)**	**RF-C4.5(*k*NN)**
1	97.06	97.06	97.06
2	93.33	100	100
3	63.16	78.94	84.21
4	73.53	76.47	88.24
5	98.66	99.33	100
6	89.36	93.62	93.62
7	98.21	97.32	98.21
8	100	99.11	100
9	100	100	100
10	64.29	83.93	89.29



Hình 9. So sánh tiêu chí Accuracy của 3 giải thuật trên 10 tập dữ liệu

V. THẢO LUẬN CÁC NGHIÊN CỨU LIÊN QUAN

Nghiên cứu chúng tôi đề xuất thay thế luật gán nhãn bình chọn số đông bởi luật cục bộ *k* láng giềng tại nút lá của giải thuật rừng ngẫu nhiên có liên quan đến các nghiên cứu trước nhằm cải tiến giải thuật học cây quyết định. Quá trình huấn luyện mô hình cây quyết định sử dụng hai chiến lược quan trọng, đó là hàm phân hoạch dữ liệu (chọn thuộc tính quan trọng, điểm phân hoạch) và luật gán nhãn ở nút lá.

Giải thuật học CART [Breiman et al., 1984], C4.5 [Quinlan, 1993] chỉ sử dụng duy nhất một thuộc tính để thực hiện phân hoạch dữ liệu. Điều này làm giảm hiệu quả phân hoạch dữ liệu do bỏ qua sự phụ thuộc của các thuộc tính trong dữ liệu. Giải thuật OC1 [Murthy et al., 1993] đề xuất xây dựng cây xiên phân nhằm kết hợp các thuộc tính để cải tiến phân hoạch dữ liệu có sự phụ thuộc lẫn nhau giữa các thuộc tính. Nghiên cứu của [Wu et al., 1999], [Do et al., 2010] đề xuất mở rộng giải thuật OC1, sử dụng máy học véctơ hỗ trợ [Vapnik, 1995], nhằm cải tiến chất lượng mô hình và tốc độ tính toán. Nghiên cứu của [Cutler & Guohua, 2001], [Geurts et al., 2006] thực hiện nhiều phân hoạch ngẫu nhiên để có mô hình tương tự như cây xiên phân.

Giải thuật Option tree của [Kohavi & Kunz, 1997] giới thiệu thêm khái niệm nút trong tùy chọn để cải thiện hiệu quả phân lớp của cây quyết định. Nghiên cứu của [Marcellin et al., 2006], [Lenca et al., 2008], [Do et al., 2010] đề xuất thay thế hàm phân hoạch (Shannon entropy) bởi entropy bất đối xứng hay khoảng cách Kolmogorov-Smirnov, nhằm cải tiến phân lớp dữ liệu không cân bằng (lớp quan tâm chiếm tỷ lệ rất ít trong tập huấn luyện so với các lớp khác).

Giải thuật Lazy tree của [Friedman et al., 96] nhằm xây dựng cây "tốt nhất" cho phần tử cần phân lớp trong pha dự đoán nhãn. Tuy nhiên luật gán nhãn tại nút lá của giải thuật cây quyết định thường dùng là luật bình chọn số đông, điều này làm giảm hiệu quả trong phân lớp. Các nghiên cứu của [Kohavi, 1996], [Seewald et al., 2000], [Pham et al., 2008] thực hiện thay thế luật gán nhãn bình chọn số đông bởi luật cục bộ như Naïve Bayes [Good, 1965] hay k láng giềng [Fix & Hodges, 1952]. Ritschard và các cộng sự đề xuất sử dụng statistical implicative analysis [Lerman et al., 1981] khi phân hoạch và gán nhãn nút lá trong giải thuật huấn luyện cây quyết định cho xử lý dữ liệu không cân bằng [Ritschard et al., 2009]. Giải thuật OK3 của [Geurts et al., 2006] sử dụng hàm nhân để thực hiện phân hoạch và gán nhãn trong giải thuật rừng ngẫu nhiên khi dự đoán cấu trúc protein.

VI. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Chúng tôi vừa trình bày đề xuất cải tiến giải thuật rừng ngẫu nhiên để nâng cao hiệu quả phân lớp các tập dữ liệu có số chiều rất lớn. Ý tưởng xuất phát từ giải thuật rừng ngẫu nhiên do Breiman đề xuất, chúng tôi đề xuất thay thế luật bình chọn số đông cho việc gán nhãn ở nút lá bằng luật quyết định cục bộ dựa vào giải thuật *k* láng giềng. Kết quả thực nghiệm trên các tập dữ liệu gen cho thấy rằng giải thuật đề xuất RF-C4.5(*k*NN) cho kết quả tốt trên tiêu chí về *Precision*, *Recall*, *F1* và độ chính xác toàn cục *Accuracy* khi so sánh với giải thuật gốc rừng ngẫu nhiên (sử dụng luật bình chọn số đông để gán nhãn ở nút lá của cây quyết định) RF-C4.5(Maj) và giải thuật máy học véctơ hỗ trợ LibSVM.

Trong tương lai, chúng tôi tiếp tục nghiên cứu các luật quyết định cục bộ dựa trên các giải thuật hiệu quả hơn *k* láng giềng. Ngoài nghiên cứu cải thiện chất lượng mô hình phân lớp, chúng tôi cũng tập trung cho cải tiến tốc độ học và phân lớp của giải thuật trong tương lai.

VII. TÀI LIỆU THAM KHẢO

[1] L. Breiman, J.H. Friedman, R.A. Olshen and C. Stone. *Classification and Regression Trees*. Wadsworth International, 1984.

[2] L. Breiman. Bagging predictors. *Machine Learning*, vol. 24, no. 2, pp. 123 - 140, 1996.

[3] L. Breiman. Arcing classifiers. *The annals of statistics*, vol. 26, no. 3, pp. 801-849, 1998.

[4] L. Breiman. Random forests. *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001.

[5] W. Buntine. Learning classification trees. *Statistics and Computing*, vol. 2, pp. 63-73, 1992.

[6] C.C. Chang and C.J. Lin. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, vol.2, no. 27, pp. 1-27, 2011. http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[7] A. Cutler and Z. Guohua. PERT – perfect random tree ensembles. *Computing Science and Statistics*, vol. 33, 2001, pp. 490-497.

[8] T-N. Do, S. Lallich, N-K. Pham and P. Lenca. Classifying very-high-dimensional data with random forests of oblique decision trees. in *Advances in Knowledge Discovery and Management*, H. Briand, F. Guillet, G. Ritschard, D. Zighed Eds, Springer-Verlag, 2010, pp. 39-55.

[9] T-N. Do, P. Lenca and S. Lallich. Enhancing network intrusion classification through the kolmogorov-smirnov splitting criterion. In: ICTACS 2010, Vietnam, 2010, pp. 50-61.

[10] E. Fix and J. Hodges. Discriminatoiry Analysis: Small Sample Performance. Technical Report 21-49-004, USAF School of Aviation Medicine, Randolph Field, USA, 1952.

[11] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Computational Learning Theory*, pp. 23-37, 1995.

[12] J-H. Friedman, R. Kohavi and Y. Yun. Lazy decision trees. In: Proceedings of the Thirteenth National Conference on Artificial Intelligence and Eighth Innovative Applications of Artificial Intelligence Conference, AAAI 96, IAAI 96, Portland, Oregon, August 4-8, 1996, vol. 1, pp. 717-724.

[13] J. Friedman, T. Hastie and R. Tibshirani. Response to Mease and Wyner, Evidence Contrary to the Statistical View of Boosting. *Journal Machine Learning Research*, vol. 9, pp. 175-180, 2008.

[14] P. Geurts, L. Wehenkel and F. d'Alché-Buc. Kernelizing the output of tree-based methods. In Cohen, W.W., Moore, A., eds.: Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, vol. 148, pp. 345-352, 2006.

[15] P. Geurts, D. Ernst and L. Wehenkel. Extremely randomized trees. *Machine Learning*, vol. 63, no. 1, pp. 3-42, 2006.

[16] I. Good. The Estimation of Probabilities: An Essay on Modern Bayesian Methods. MIT Press, 1965.

[17] A.J. Grove and D. Schuurmans. Boosting in the limit: Maximizing the margin of learned ensembles. In Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI-98), 1998, pp. 692-699.

[18] J. Han, M. Kamber and J. Pei. *Data Mining: Concepts and Techniques*. Morgan Kaufmann; 3 edition, 2011.

[19] L. Jinyan and L. Huiqing. Kent ridge bio-medical dataset repository. 2002, http://datam.i2r.astar.edu.sg/-datasets/krbd/.

[20] R. Kohavi and C. Kunz. Option decision trees with majority votes. In: Proceedings of the Fourteenth International Conference on Machine Learning (ICML 1997), Nashville, Tennessee, USA, July 8-12, 1997, pp. 161-169.

[21] R. Kohavi. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA, 1996, pp. 202-207.

[22] P. Lenca, S. Lallich, T-N. Do and N-K. Pham. A comparison of different off-centered entropies to deal with class imbalance for decision trees. In: The Pacific-Asia Conference on Knowledge Discovery and Data Mining, LNAI 5012, Springer-Verlag, 2008, pp. 634-643.

[23] I. Lerman, R. Gras and H. Rostam. Elaboration et évaluation d'un indice d'implication pour données binaires. *Mathématiques et Sciences Humaines*, vol. 74, pp. 5-35, 1981.

[24] S. Marcellin, D. Zighed and G. Ritschard. An asymmetric entropy measure for decision trees. In: IPMU 2006, Paris, France, 2006, pp. 1292-1299.

[25] S. Murthy, S., Kasif, S., Salzberg, R., Beigel. OC1: Randomized induction of oblique decision trees. In: Proceedings of the Eleventh National Conference on Artificial Intelligence, 1993, pp. 322-327.

[26] N-K. Pham, T-N. Do, P. Lenca and S. Lallich. Using local node information in decision trees: coupling a local decision rule with an off-centered. In: International Conference on Data Mining, Las Vegas, Nevada, USA, CSREA Press, 2008, pp. 117-123.

[27] J.R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.

[28] G. Ritschard, S. Marcellin and D. Zighed. Arbre de décision pour données déséquilibrées: sur la complémentarité de l'intentisé d'implication et de l'entropie décentrée. In: Analyse Statistique Implicative - Une méthode d'analyse de données pour la recherche de causalités, 2009, pp. 207-222.

[29] A-K. Seewald, J. Petrak and G. Widmer. Hybrid decision tree learners with alternative leaf classifiers: An empirical study. In: Int. Florida Artificial Intelligence Research Society Conference, 2000, pp. 407-411.

[30] C.V. van Rijsbergen. *Information Retrieval*. Butterworth, 1979.

[31] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1995.

[32] D. Wolpert. Stacked generalization. *Neural Networks*, vol. 5, pp. 241-259, 1992.

[33] W. Wu, K. Bennett, N. Cristianini and J. Shawe-Taylor. Large margin trees for induction and transduction. In: Proceedings of the Sixth International Conference on Machine Learning, 1999, pp. 474-483.

[34] Q. Yang and X. Wu. 10 Challenging Problems in Data Mining Research. *Journal of Information Technology & Decision Making*, vol. 5, no. 4, pp. 597-604, 2006.

RANDOM FORESTS USING LOCAL LABELING RULES FOR IMPROVING CLASSIFICATION CORRECTNESS

Thanh-Nghi Do, Nguyen-Khang Pham, Huu-Hoa Nguyen, Minh-Trung Nguyen

ABSTRACT *- In this paper, we propose to use local labeling rules in random forests of decision trees for effectively classifying data. The decision rules use the majority vote for labeling at terminal nodes in decision trees, maybe making the classical random forest algorithm degrade the classification performance. Our investigation aims at replacing the majority rules with the local ones, i.e. k nearest neighbors to improve the prediction correctness of decision forests. The numerical test results on gene datasets from datam.i2r.a-star.edu.sg/datasets/krbd showed that that our proposal gives good classification results compared with classical random forests and support vector machine (SVM) in terms of Precision, Recall, F1 and Accuracy.*