BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐH SƯ PHẠM KĨ THUẬT TP HỒ CHÍ MINH

KHOA ĐÀO TẠO CHẤT LƯỢNG CAO

BÁO CÁO ĐỒ ÁN 2

ĐỀ TÀI: TÌM HIỂU VỀ BOOSTING VÀ THUẬT TOÁN ADABOOST

Sinh viên thực hiện: Lê Bá Huỳnh 16110095

Ngô Thanh Tài 16110201

Giảng viên hướng dẫn: Thầy Trần Nhật Quang

Thời gian thực hiện từ ngày 23/4/2019-23/5/2019

MỤC LỤC

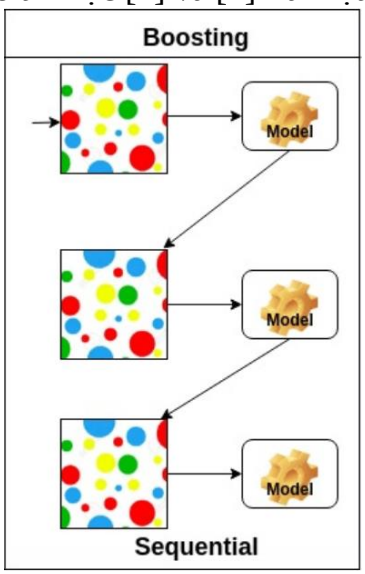
DANH MỤC CÁC HÌNH

Phần 1: GIỚI THIỆU

1. *Boosting và thuật toán Adaboost*

1.1 Boosting

-Là một kỹ thuật sử dụng các bộ phân lớp yếu và tổng hợp lại thành một bộ phân lớp mạnh. Kỹ thuật này được thực hiện bằng việc xây dựng ra mô hình từ dữ liệu ban đầu, sau đó tạo ra mô hình tiếp theo bằng việc sửa lỗi từ mô hình trước, và cứ tuần tuần tự như thế. Cuối cùng cho ra là một mô hình được dự đoán là hoàn hảo. Tham khảo từ mục [1] và [2] Tài liệu tham khảo.



Hình 1: Boosting - Tham khảo từ mục [1] Tài liệu tham khảo.

1.2 Thuật toán Adaboost

- Tham khảo từ mục [2] và [4] Tài liệu tham khảo.

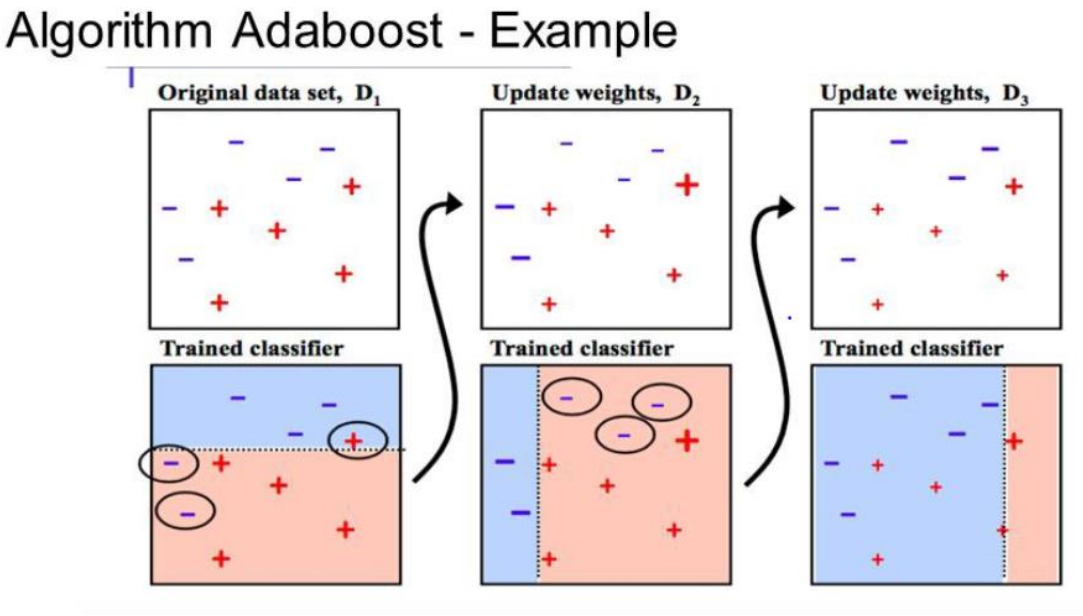
- Là một thuật toán áp dụng kỹ thuật boosting. Được phát minh vào năm 1996 bởi Yoav Freund và Robert Schapire là hai nhà khoa học máy tính, chuyên về học máy. Và với thuật toán này, cả hai đã nhận được giải thưởng Gödel Prize vào năm 2003.

- Adaboost được coi là thuật toán được phát triển thành công trong việc phân loại nhị phân và được coi là bắt đầu tốt nhất khi tìm hiểu về boosting.

- Adaboost có rất nhiều ứng dụng trong thực tế, được áp dụng cụ thể vào các bài toán nhận diện như là nhận diện khuôn mặt, các chữ số, biển số…

- Đặc điểm của thuật toán này là gán cho mỗi một phân lớp một trọng số.

Ở mỗi vòng lặp ta sẽ tiến hành cập nhật trọng số cho bộ phân lớp yếu vừa xây dựng bằng cách tăng trọng số của mẫu phân lớp sai, giảm trọng số của mẫu phân lớp đúng. Và sau đó kết hợp tuyến tính các bộ phân lớp yếu lại với nhau để cho ra một bộ phân lớp mạnh.



Hình 2: Ví dụ về thuật toán Adaboost mục [1] Tài liệu tham khảo

- Điểm quan trọng là các dữ liệu của bạn phải rõ ràng ở các giá trị biên. Bởi vì Adaboost nhạy cảm với một số dữ liệu.

2. *Mục tiêu đề ra*

- Hiểu được vấn đề đặt ra trong thuật toán Adaboost. Sau đó xây dựng hoặc sử dụng thư viện để chạy thử thuật toán, cho dữ liệu vào thuật toán để cho ra kết quả tối ưu. Xem các kết quả đạt được như thế nào để rồi rút ra kết luận về thuật toán.

3. *Phạm vi đề tài*

- Adaboost là một thuật toán áp dụng kỹ thuật boosting trong bộ môn Machine Learning(Học máy).

- Thuật toán có thể được áp dụng rộng rãi trong nhiều ứng dụng thực tế.

Phần 2: MÔ TẢ, PHÂN TÍCH THUẬT TOÁN ADABOOST

Cho tập ảnh huấn luyện $(x_1, t_1), \ldots, (x_n, t_n)$ với $t_i \in \{0, 1\}$

Bước 1: Ta khởi tạo trọng số cho mỗi mẫu huấn luyện: $w_n^{(1)} = 1/N$ với n = 1, 2, …, N là bằng nhau, nó thể hiện độ quan trọng của mẫu huấn luyện đó.

Bước 2: Tạo vòng lặp For m = 1, …, M

- Trong mỗi vòng lặp ta thực hiện 2 việc:

Thứ nhất: Xây dựng bộ phân lớp yếu h_m:

+ Với mỗi đặc trưng j, xây dựng một bộ phân lớp h_j với độ lỗi:

$$E_j = \sum_{n=1}^{N} w_n^{(m)} I(h_m(x_n) \neq t_n)$$

với $I(h_m(x_n) \neq t_n) = 1$ nếu $h_m(x_n) \neq t_n$ và = 0 nếu ngược lại.

+ Chọn bộ phân lớp h_j với độ lỗi nhỏ nhất ta được h_m.

Thứ hai: Cập nhật trọng số:

+ Tính:

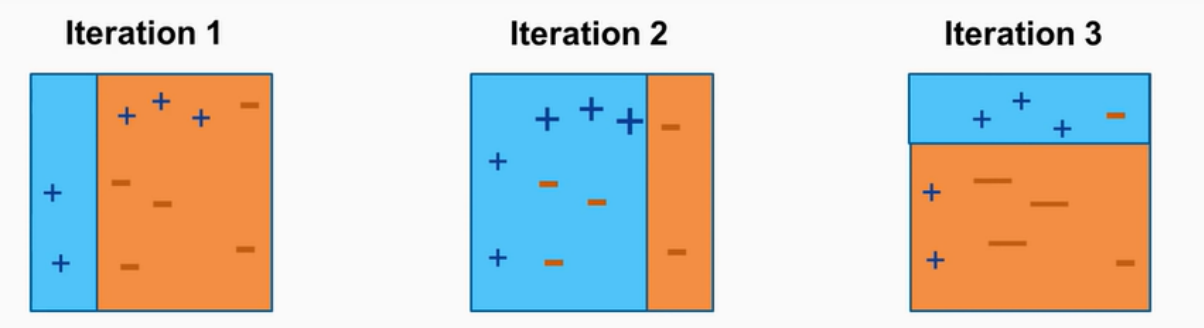
$$\epsilon_m = \frac{\sum_{n=1}^{N} w_n^{(m)} I(h_m(x_n) \neq t_n)}{\sum_{n=1}^{N} w_n^{(m)}}$$

và:

$$\alpha_m = ln \frac{1 - \epsilon_m}{\epsilon_m}$$

+ Cập nhật trọng số:

$$w_n^{(m+1)} = w_n^{(m)} \exp\{\alpha_m I(h_m(x_n) \neq t_n)\}$$



Hình 3: Phân loại các lớp yếu mục [3] Tài liệu tham khảo

Bước 3: Kết hợp tuyến tính lại ta được bộ phân lớp mạnh cuối cùng:

$$H_M(x) = sign[\sum_{m=1}^{M} \alpha_m h_m(x)]$$

Hình 4: Kết quả một lớp mạnh mục [3] Tài liệu tham khảo

Phần 3: XÂY DỰNG, KIỂM THỬ THUẬT TOÁN ADABOOST

1. *Lập trình*

- Tham khảo từ mục [1] Tài liệu tham khảo

- Import các thư viện cần thiết

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.colors import ListedColormap
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import AdaBoostClassifier
```

Hình 5: Import thư viện

- Đọc và phân tách dữ liệu thành các features và label tương ứng.

```
df = pd.read_csv("adaboost.csv")
X = df.values[:, :2] #lay tat ca gia tri 2 cot dau tien
y = df.Decision
```

Hình 6: Code đọc dữ liệu

- Tách dữ liệu thành 2 phần Train(70%) và Test(30%)

```
# Split dataset into training set and test set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
```

Hình 7: Code tách dữ liệu

- Tạo list màu hiển thị cho các vùng dữ liệu

```
# Create color maps
cmap_light = ListedColormap(['#FFAAAA', '#AAFFAA', '#AAAAFF'])
cmap_bold = ListedColormap(['#FF0000', '#00FF00', '#0000FF'])#
```

Hình 8:Tạo màu cho phân vùng dữ liệu

- Tạo Object Adaboost Classifer và train dữ liệu

```
# Create adaboost classifer object
abc = AdaBoostClassifier(n_estimators=200,
                         learning_rate=1)
# Train Adaboost Classifer
model = abc.fit(X_train, y_train)
```

Hình 9: Tạo đối tượng Adaboost và train dữ liệu

- Căn chỉnh giới hạn hiển thị của biểu đồ theo bộ dữ liệu

```
#can chinh gioi han hien thi cua bieu do
x_min = X[:, 0].min() -1
x_max = X[:, 0].max() + 1
y_min = X[:, 1].min() - 1
y_max = X[:, 1].max() + 1
```

Hình 10: Code chỉnh giới hạn của biểu đồ

- Thực hiện tính toán và hiển thị phân vùng dữ liệu

```
xx, yy = np.meshgrid(np.arange(x_min, x_max, 0.02),np.arange(y_min, y_max, 0.02))
Z = model.predict(np.c_[xx.ravel(), yy.ravel()])
# Put the result into a color plot
Z = Z.reshape(xx.shape)
plt.pcolormesh(xx, yy, Z, cmap=cmap_light)#hien thi phan vung


# Plot also the training points
plt.scatter(X[:, 0], X[:, 1], c=y, cmap=cmap_bold)
#X[:, 0]: tat ca cac gia tri cua cot thu 0 trong X
#X[:, 1]: tat ca cac gia tri cua cot thu 1 trong X

plt.xlim(xx.min(), xx.max())
plt.ylim(yy.min(), yy.max())
plt.title("Adaboost")
plt.show()
```

Hình 11: Code thực hiện và vẽ phân vùng dữ liệu

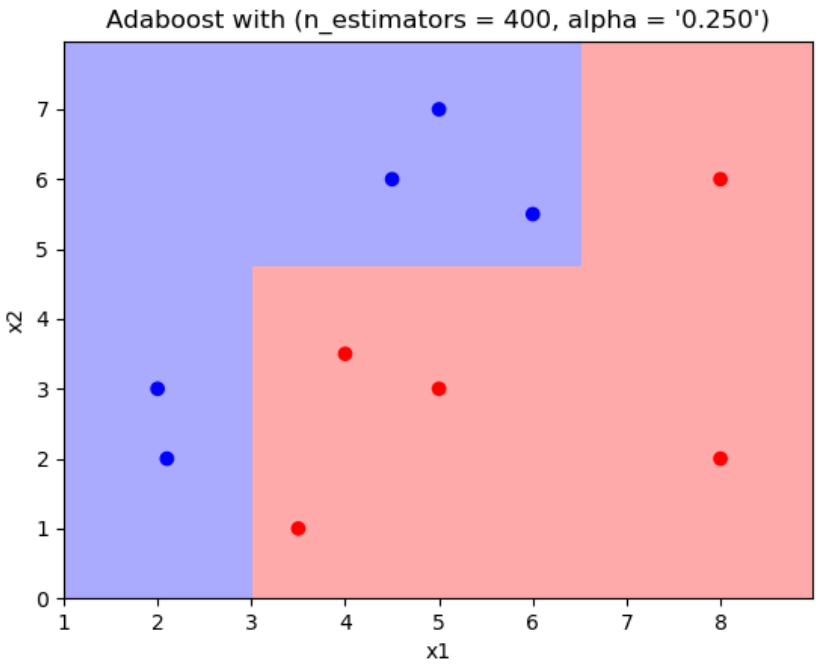
2. *Dữ liệu*

- Dữ liệu với 2 features là x1 và x2, label là Decision

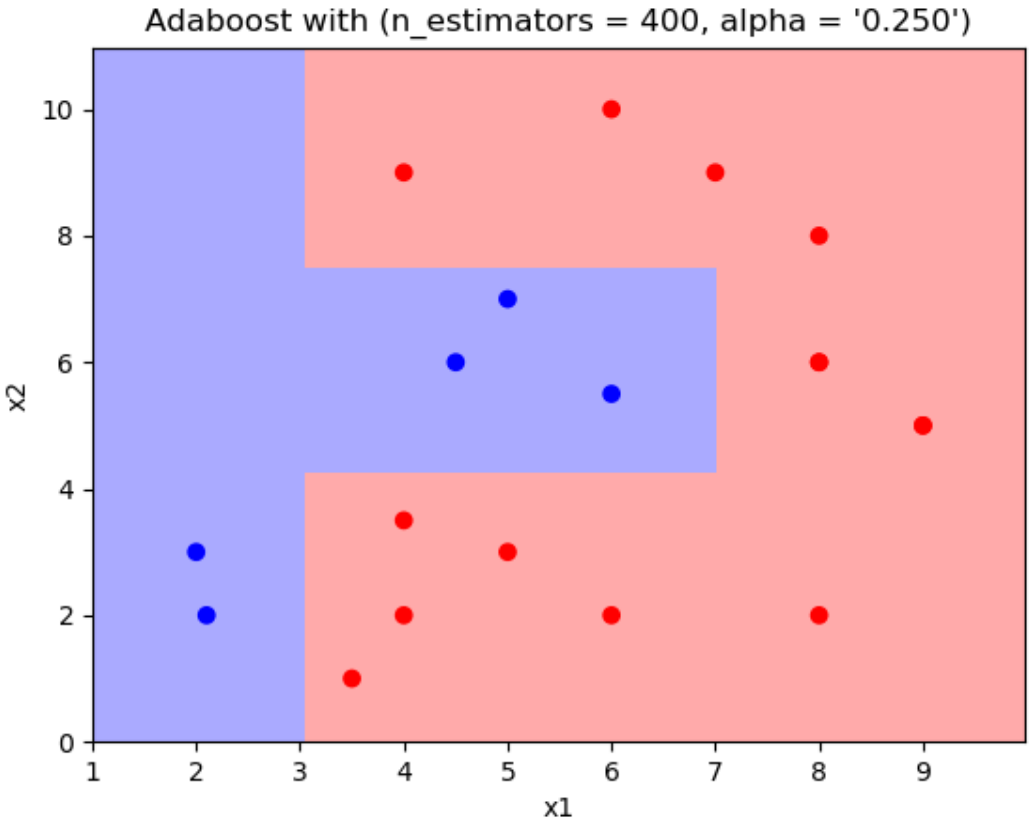
x1	x2	Decision
14	4	-1
19	16	-1
5	15	-1
16	11	-1
13	2	-1
2	8	-1
2	19	-1
10	12	-1
4	11	-1
7	11	-1
7	3	-1
15	17	-1
14	16	-1
13	3	-1
3	9	-1
11	14	-1
7	7	-1
10	7	-1
15	5	-1
11	6	-1
10	14	-1

Hình 12: Dữ liệu

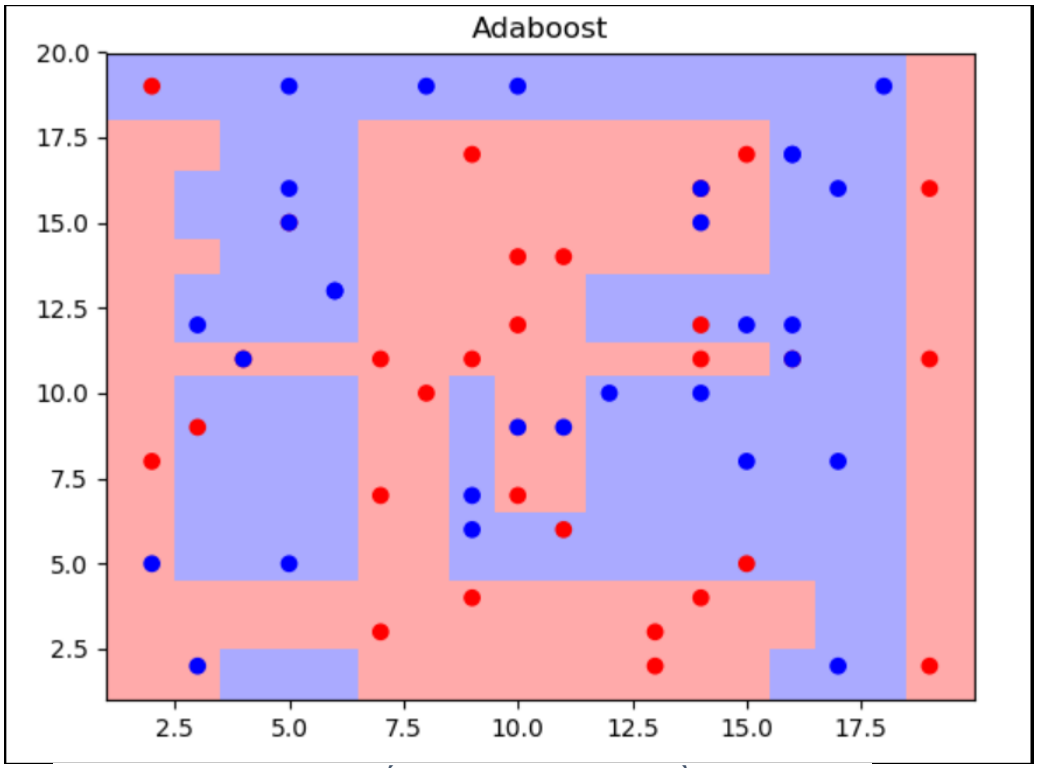
3. *Kết quả sau khi chạy thuật toán*



Hình 13: Kết quả với dữ liệu ít



Hình 15: Kết quả với dữ liệu vừa



Hình 14: Kết quả với dữ liệu nhiều

Phần 4: KẾT LUẬN

1. *Mức độ hoàn thành*

- Đã tìm hiểu về Boosting và thuật toán Adaboost.
- Sử dụng được thư viện của python để chạy thuật toán.
    + Chưa tự cài đặt được thuật toán.
- Vẽ được biểu đồ dữ liệu từ các tập tin dữ liệu.
- Chạy thử được dữ liệu tự tạo và dữ liệu có sẵn.

2. *Các khó khăn gặp phải*

- Vẫn chưa cài đặt hoàn toàn thuật toán
    + Cách khắc phục: Sử dụng thư viện sklearn để chạy thuật toán.
- Vẽ biểu đồ từ các tập tin dữ liệu.
    + Cách khắc phục: Phải tham khảo cách vẽ từ nhiều nguồn.

3. *Ưu, nhược điểm của thuật toán*

- Ưu điểm:
    + Phân lớp được các dữ liệu phức tạp một cách rõ ràng.
    + Áp dụng được các thuật toán khác.
- Nhược điểm:
    + Nhạy cảm với noisy data and outliers
    + Dữ liệu phải có giá trị biên rõ ràng.
    + Số lần học của thuật toán càng lớn chạy cằng lâu.

4. *Ý tưởng phát triển*

- Có thể sử dụng thuật toán để làm các dự án nhỏ như nhận diện các con số viết tay từ 0 đến 9 hoặc bảng chữ cái. Và cao hơn thì có thể áp dụng vào việc nhận dạng biển số xe, khuôn mặt,…

Phần 5: TÀI LIỆU THAM KHẢO

[1]https://www.datacamp.com/community/tutorials/adaboost-classifier-python?fbclid=IwAR34h1TMn-25uXRLjnGEN0UWG1BX79d0QXeBXczF1BLA1XazX-qblUTfKAo

[2] https://techtalk.vn/top-10-thuat-toan-machine-learning-danh-cho-newbie.html

[3] https://www.youtube.com/watch?v=BoGNyWW9-mE

[4] https://en.wikipedia.org/wiki/AdaBoost

Phần 6: BẢNG PHÂN CÔNG VIỆC

STT	Công việc	Huỳnh	Tài
1	Tìm hiểu coi Boosting và thuật toán Adaboost là gì?	×	×
2	Tìm hiểu về công thức của thuật toán Adaboost và môi trường xây dựng thuật toán	×	×
3	Xây dựng thuật toán	×	×
4	Tìm kiếm dữ liệu cho thuật toán	×	×
5	Áp dụng dữ liệu vào thuật toán và kiểm thử thuật toán	×	×
6	Viết báo cáo	×	×

Phần 7: ĐẠO VĂN LÀ GÌ?

- Đạo văn được coi là một hành động ăn cắp hay sao chép lại ý tưởng, ngôn từ của người khác và coi nó như của mình một cách công khai và hiển nhiên. Nó chưa có khái niệm hay định nghĩa một cách cụ thể. Chỉ được coi là hành vi gian lận, thiếu trung thực, và mang lại nhiều ảnh hưởng xấu cho người đó. Người đạo văn đôi khi sẽ bị mất kiến thức về mảng kiến thức đó, không những thế còn dẫn tới các hệ lụy như: Vi phạm bản quyền đối với các sản phẩm có sự công nhận của Pháp luật, gian lận trong thi cử đối các bài copy. Nó ảnh hưởng rất nhiều về mặt nhân phẩm và đạo đức.

- Tuy nhiên có một số ý kiến lại cho rằng đạo văn chỉ là vay mượn ý tưởng của người khác nhằm phát huy cho nó tốt hơn, mới mẻ hơn. Và để tránh việc đạo văn tham khảo tại mục [13] Tài liệu tham khảo thì các bạn nên:

+ Tìm hiểu rõ vấn đề của bạn và diễn đạt bằng từ ngữ của chính mình.

+ Ghi rõ nguồn được trích từ đâu.

+ Hiểu rõ về vi phạm bản quyền vì nó khá là nguy hiểm.

+ Nên trích dẫn đúng cách và đúng phần.

+ Phải hiểu rõ vấn đề được nêu ở trong nguồn.

-Các bạn không nên:

+ Sao chép ngôn từ, lời lẽ của người khác.

+ Lấy đoạn trích mà không ghi nguồn.

+ Đưa thông tin sai lệch khi trích nguồn.

"Chúng tôi xin cam đoan đồ án này do chính chúng tôi thực hiện. Chúng tôi không sao chép, sử dụng tài liệu, mã nguồn của người khác mà không ghi rõ nguồn gốc. Chúng tôi xin chịu hoàn toàn trách nhiệm nếu vi phạm"

Lê Bá Huỳnh

Ngô Thanh Tài