Báo cáo tìm hiểu đề tài cuối kỳ

Mục Lục

[I. Giới thiệu về đề tài : 2](#_Toc294023548)

[II. Các phương pháp ẩn dữ liệu trong báo cáo 2](#_Toc294023549)

[o NICETEXT 2](#_Toc294023550)

[o TEXTTO 2](#_Toc294023551)

[o MARKOV CHAIN 2](#_Toc294023552)

[III. Word Location: 3](#_Toc294023553)

[Word location: 3](#_Toc294023554)

[Sự khác biệt về SD trong văn bản tự nhiên và văn bản nhúng 4](#_Toc294023555)

[IV. Support Vector Machine 4](#_Toc294023556)

[o Khái niệm 4](#_Toc294023557)

[o Vai trò SVM trong bài toán này 6](#_Toc294023558)

[o Mô hình huấn luyện sử dụng SVM 6](#_Toc294023559)

[V. Huấn luyện và kiểm tra 7](#_Toc294023560)

[o Đặc tả dữ liệu 7](#_Toc294023561)

[o Kết quả huấn luyện và kiểm tra 7](#_Toc294023562)

[Kết quả 7](#_Toc294023563)

[Sơ đồ phân loại 8](#_Toc294023564)

[o Nhận xét 9](#_Toc294023565)

# Giới thiệu về đề tài :

Đề tài của nhóm là tìm hiểu về phương pháp kiểm tra văn bản có ẩn dữ liệu hay không bằng Word Location . Các kỹ thuật ản dữ liệu được kiểm tra là NICETEXT, TEXTTO và Markov Chain. Từ phương pháp này , một machine sẽ được tạo ra và huấn luyện để phân biệt văn bản có nhúng bằng ba phương pháp nhúng ở trên không ?

Danh sách các thành viên nhóm 5.

|  |  |  |
| --- | --- | --- |
| Tên | MSSV | Đóng góp (%) |
| Nguyễn Xuân Huy | 0712196 | 50 |
| Trần Văn Tiến | 0712446 | 50 |
| Diệp Thế Nghĩa | 0712301 | 0 |

# Các phương pháp ẩn dữ liệu trong báo cáo

## NICETEXT

|  |  |  |  |
| --- | --- | --- | --- |
| **Style** | **Type** | **Bit** | **Word** |
| Footballer Fb\_Verb Fb\_Object | Footballer | 0 | Striker |
| 1 | Middlefier |
| Fb\_Verb | 0 | take |
| 1 | stop |
| Fb\_Object | 0 | a pass |
| 1 | a shot |

## TEXTTO

|  |  |  |
| --- | --- | --- |
| **TYPE** | **Index** | **Word** |
| THING | 21 | referre |
| ADVERB | 31 | wrongly |
| VERB | 30 | decide |
| ADJECTIVE | 18 | stupid |
| PLACE | 30 | stadium |

iluvu -> 18 21 30 31 30 -> The stupid referre decide wrongly in the stadium.

## MARKOV CHAIN

# Word Location:

### Word location:

Trong phương pháp này, chúng ta xem các câu liền kề nhau như là 1 đoạn văn bản. Vậy chúng ta có thể xem một phần của văn bản hay toàn bộ văn bản như 1 phân đoạn. Trong thí nghiệm này. Tất cả các công việc ta sẽ xét trên các phân đoạn văn bản.

Một văn bản được chia thành nhiều phân đoạn. Trong mỗi phân đoạn ta sẽ có các từ, và thứ tự của nó.



Với Wi, 0<= i<= n-1, là thứ tự của từ trong phân đoạn.

Vị trí của từ trong văn bản được xác định



Dể dàng nhận thấy rằng, 0< WLi<1.

Thực ra , trong 1 phân đoạn văn bản. Chỉ có xác định một số hữu hạn m các từ ( Wk’ ) và mỗi từ thì có nk hiển thị ( lúc chổ này lúc chổ khác). Do đó ta có 1 phép tính.



Nghĩa là: Số từ có trong đoạn văn bản, sẽ bằng tổng tập hợp các từ có trong văn bản đó nhân với số lần xuất hiện của chính nó ( tôi hiểu vậy ).

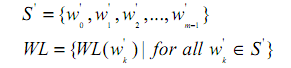
Đối với 1 từ Wk thì có nk hiển thị. Do đó ta nói từ Wk có một set các words location.

1 set vị trí của từ Wk  được biểu thị bằng LS ( Wk)



Đặt S’ là tập hợp các từ riêng biệt có trong Văn bản. WL là tập hợp địa chỉ của chúng.

Ta có công thức.

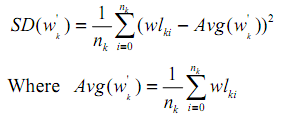


Và đây là dạng khác của biểu diển 1 đoạn văn bản.



1. Statistical Variables of Word Location ( thống kê biến đổi WL).

Trong văn bản tự nhiên. Sự lặp lại của một từ luôn thể hiện sự phân bổ không đồng đều. Một từ có thể xuất hiện nhiều ở một đoạn nào đó, nhưng ở chổ khác thì hiếm khi xuất hiện. Đặt SD ( chỉ số dàn trải : spread degree) để biểu thị điều đó.



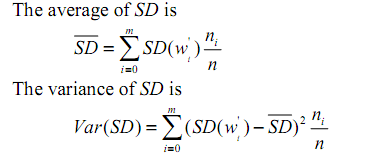
Từ công thức của một đoạn văn bản



Cho thấy,

Mỗi phân đoạn văn bản sẽ là tập hợp của một số từ ( dĩ nhiên là mỗi từ sẽ có 1 tập hợp vị trí của nó)

Chúng ta có thể do được sự phân bổ của từ trong một phân đoạn văn bản bằng cách kiểm tra sự phân bổ của chỉ số SD.



Giá trị SD trung bình và VarSD biểu thị đặc trưng phân bổ của 1 từ xác định. Do đó có thể dùng nó như 1 luật để phân loại văn bản thường và văn bản đã có ẩn dữ liệu.

<có nên cho ví dụ về 1 đoạn văn bản cụ thể hay không ? >

### Sự khác biệt về SD trong văn bản tự nhiên và văn bản nhúng

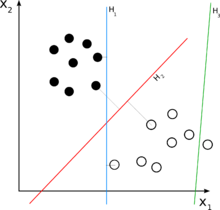
# Support Vector Machine

## Khái niệm

Support vector machines (SVMs) là phương pháp học giám sát gồm phân tích dữ liệu và nhận dạng mẫu.. Thuật toán SVM đầu tiên được Vladimir Vapnik giới thiệu và ông cùng Corinna Cortes đã đề xuất chuẩn SVM đang được sử dụng hiện nay.

SVM là phương pháp phân loại tuyến tính. Các mẫu sẽ được phân loại dựa trên các tính chất khác nhau của mẫu đó. Ví dụ một để phân loại cho các đối tượng thuộc một trong hai lớp khác nhau có n tính chất dùng để phân loại. Ta thể hiện n tính chất này thành trong không gian vector n chiều và tìm một siêu phẳng n-1 chiều để phân tách thành 2 lớp khác nhau.

Ví dụ có 2 lớp Táo và Lê, các tính chất dùng để phân biệt quả táo hay quả lê là chiều dài và cân nặng. Ta lấy thể hiện các tính chất này lên đồ thị 2 chiều thì được hình dưới. Với x1 thể hiện chiều dài ,x2 thể hiện cân nặng . Các điểm trắng thể hiện các quả lê trong khi các điểm đen là các quả táo.



Sau đó ta tìm một đường thẳng để tách 2 điểm táo và lê. Đường thẳng này được gọi là siêu phằng 1 chiều. Do có nhiều đường thẳng hợp phù hợp nên ta chon đường thẳng các xa cách xa các điểm trằng và đen nhất. Trong đồ thị trên thì đường phân loại là đường H2 màu đỏ .Khi đó nếu một quả có chiều cao là x1 và cân nặng là x2 nếu điểm (x1,x2) nằm bên trái đường thẳng H2 trên thì nó là quả táo , ngược lại nó là quả lê.

Trong trường hợp đối tượng được xét không có tính chất biến đổi tuyến tính. Ví dụ như các điểm táo và lê trong hính 1 nằm xen kẽ nhau không vẽ đường phân cách được ta phải dùng hàm kernel để mở rộng số chiêu không gian tính chất .Khi đó ta có thể khảo sát các tính chất chiều dài và cân nặng trong không gian 3 chiều hoặc 4 chiều.

Quá trình sử dụng SVM được chia làm 2 bước .

* Bước huấn luyện ta phải đưa dữ liệu để nó tìm để nó tìm hàm phân loại tốt nhất. Ví dụ ta đưa một số quả táo và lê với chiều dài và cân nặng của chúng cho SVM. SVM sẽ thể hiện đồ thị tính chất như ở trên và tìm đường thẳng để phân loại.
* Bước kiểm tra người ta chỉ đưa các thuộc tính và máy phải phân biệt nó thuộc nhãn nào. Tỉ lệ phân loại chính xác của máy là cơ sở để xem phương pháp SVM có thành công không . Trong ví dụ trên ta sẽ đưa một quả biết chiều dài và cân nặng để SVM phân loại xem nó là quả táo hay quả lê. Sau một số lần phân loại của SVM ta sẽ tính được xác suất phân loại thành công của SVM. Nếu xác suất này thấp ta phải kiểm tra và bổ sung thêm các tính chất được dùng để phân loại như màu sắc, độ bóng.

So với các phương pháp phân loại khác như cây quyết định và phân loại Bayes thì SVM có thể xứ lý trong đối tượng có nhiều tính chất hơn do sử dụng không gian n chiều và siêu phẳng. Đồng thời SVM cũng khử nhiễu tốt hơn

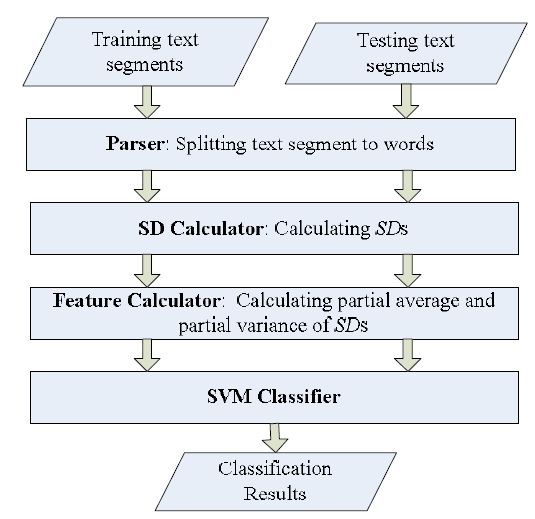
Phương pháp SVM có cài đặt sẵn trong MATLAB . Nên ta dễ dàng thực hiện việc huấn luyện và kiểm tra theo SVM.

## Vai trò SVM trong bài toán này

Trong bài toán trên ta cần phân biệt hai lớp là dữ liệu không nhúng (được gọi là dữ liệu tốt) và dữ liệu có nhúng watermark (gọi là dữ liệu xấu). Các tính chất dùng để phân loại là và. Do chỉ tính được các tính chất này trong văn bản mà chưa có kiến thức nào về mối quan hệ giữa các tính chất với dữ liệu tốt hay dữ liệu xấu nên nểu không dùng các phương pháp máy học như SVM thì ta. Nếu SVM phân loại thành công (độ chính xác khi kiểm tra đạt khoảng 85% trở lên) thì ta có thể khẳng định có thể phân loại dữ liệu tốt và dữ liệu xấu nhờ vào các tính chất này . Ngược lại các tính chất chưa đủ để kiểm tra xem văn bản có đã được nhúng watermark bên trong hay không ?

## Mô hình huấn luyện sử dụng SVM

Mô hình huấn luyện sử dụng SVM để phát hiện stego text được thể hiện như sau



Mô hình này gồm có hai pha là huấn luyện và kiểm tra. Dữ liệu đầu vào “thô”của mỗi pha là segment text. Segment text là một đoạn văn bản nhỏ có thể là một văn bản hoàn chỉnh hay là một phẩn của văn bản hoàn chỉnh. Trong mô hình trên các segment text sẽ được rút ra các giá trị varSD và avgSD làm dữ liệu để huấn luyện và kiểm tra. Các segment text được chia làm 2 loại training và testing

Trong pha huấn luyện SVM Classifier sẽ thu thập các thông tin về varSD và avgSD của training segment text để tạo ra hàm phân loại. Hàm phân loại này sẽ được sử dụng trong pha kiểm tra nhằm phân loại testing segment text và đánh giá độ thành công. Tùy vào độ thành công trong pha kiểm tra mà ta đánh giá phương pháp phân loại văn bản tự nhiên và stego text dựa trên varSD và avgSD có đáng tin cậy hay không ?

# Huấn luyện và kiểm tra

## Đặc tả dữ liệu

Trong quá trình thực nghiệm phương pháp phát hiện mã hóa này chúng em dùng hai loại dữ liệu là good data và bad data. Good data là dữ liệu văn bản chữ viết tự nhiên , Bad data là dữ liệu văn bản được tạo từ phương pháp TEXTO. Phần thực nghiệm này được tiến hành trên 3 loại kích thước dữ liệu khác nhau là 5KB, 10KB và 20KB với số lượng được cho trong bảng sau.

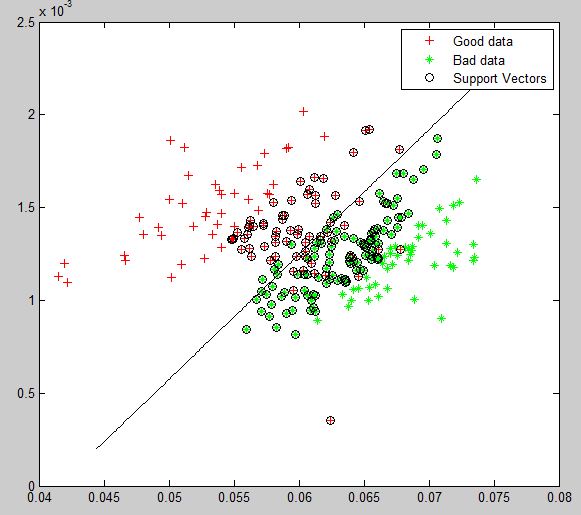
|  |  |  |  |
| --- | --- | --- | --- |
| **Kích thước** | **Mode** | **Kiểu dữ liệu** | **Số lượng file** |
| 5KB | Huấn luyện | Good Data | 109 |
| Bad Data | 152 |
| Kiểm tra | Good Data | 100 |
| Bad Data | 100 |
| 10KB | Huấn luyện | Good Data | 150 |
| Bad Data | 150 |
| Kiểm tra | Good Data | 100 |
| Bad Data | 100 |
| 20KB | Huấn luyện | Good Data | 136 |
| Bad Data | 150 |
| Kiểm tra | Good Data | 100 |
| Bad Data | 100 |

## Kết quả huấn luyện và kiểm tra

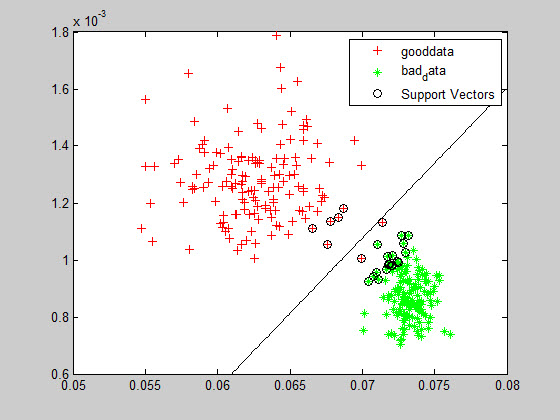
### Kết quả

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kích thước** | **Kiểu dữ liệu** | **Thành công** | **Thất bại** | **Tỉ lệ** | **Tổng cộng** |
| 5KB | Good data | 88 | 12 | 88% | 91.5% |
| Bad data | 95 | 5 | 95% |
| 10KB | Good data | 99 | 1 | 99% | 99.5% |
| Bad data | 100 | 100 | 100% |
| 20KB | Good data | 99 | 1 | 99% | 99.5% |
| Bad data | 100 | 100 | 100% |

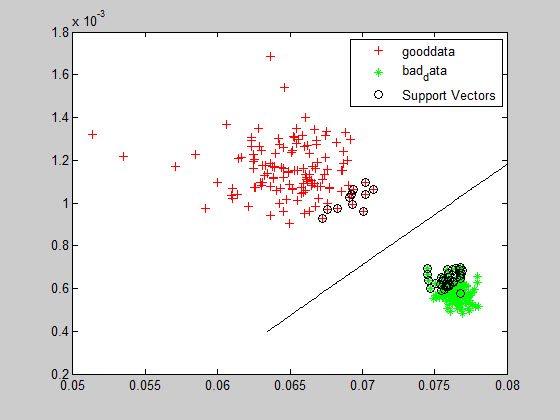
### Sơ đồ phân loại



Hình 1. Kết quả huấn luyện bằng SVM với dữ liệu 5KB



Hình 2. Kết quả huấn luyện bằng SVM với dữ liệu 10KB



Hình 3. Kết quả huấn luyện bằng SVM với dữ liệu 20KB

## Nhận xét

Kết quả huấn luyện và kiểm tra cho kết quả thành công cao điều này cho thấy ta cỏ thể phân biệt dữ liệu văn bản tự nhiên và dữ liệu bị nhúng bởi phương pháp TEXTO nhờ vào var(SD) và avg(SD).

Kết quả thực nghiệm của nhóm em cao hơn kết quả thực nghiệm của bài báo cáo có thể vì do cách bỏ qua các giá trị n nhỏ. Trong bảng Figure 2 trang 560 của paper thì giá trị n’ khác với n do bỏ qua các từ xuất hiện quá ít nhưng không cho biết là xuất hiện bao nhiêu lần thì được xem là ít. Vì thế nhóm em chọn bỏ qua các từ chỉ xuất hiện một lần nên cách tính var(SD) và avg(SD) có thể khác biệt một ít với phương pháp của tác giả. Một nguyên nhân khác là phương pháp nhúng của nhóm chỉ là TEXTTO thay vì cà ba phương pháp như trong bài báo cáo do không tìm thấy các source code của các phương pháp kia . Do chỉ một phương pháp nhúng nên các Bad Data sẽ phân bố trong vùng hẹp hơn.

Trong ba ảnh phân loại trên ta thấy rằng dữ liệu có kích thước càng lớn thì càng dễ huấn luyện và detect. Nguyên nhân là do khi các từ xuất hiện càng nhiểu thì sự khác biệt giữa văn bản viết và văn bản nhúng càng rõ ràng.

Một điều đáng lưu ý là khi văn bản nhúng có kích thước càng cao thì var(SD) và avg(SD) càng ít thay đổi. Vì trong từ điển của TEXTTO xác suất xuất hiện của các từ trong cùng một type luôn bằng nhau. Khi dữ liệu càng lớn thì xác suất này càng thể hiện rõ nên các giá trị SD của các từ sẽ càng ổn định hơn dẫn đến var(SD) và avg(SD) ít biến đổi.

# Tổng kết