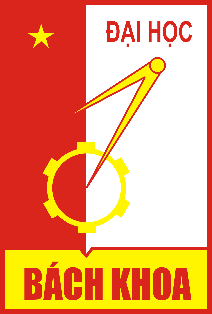
**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

──────── \* ───────



**BÁO CÁO MÔN HỌC**

**Xử lí ngôn ngữ tự nhiên**

**Tên đề tài: Phân loại văn bản bằng thuật toán học**

**có giám sát**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | **PGS.TS.LÊ THANH HƯƠNG** |
| **Sinh viên thực hiện:** | **Nhóm 15** |
| Hoàng Mạnh Hiệp | 20161438 |
| Trần Thanh Tú | 20164487 |
| Ngô Minh Hải | 20179569 |
|  |  |

**HÀ NỘI 12-2019**

Mục Lục

[Chương 1 Phát biểu bài toán 3](#_heading=h.30j0zll)

[Chương 2: Tiền xử lý dữ liệu 4](#_heading=h.3znysh7)

[2.1 Các bước tiến hành. 4](#_heading=h.2et92p0)

[2.2 Ưu điểm và nhược điểm của cách tiền xử lý dữ liệu 5](#_heading=h.tyjcwt)

[Chương 3: Lựa chọn mô hình 6](#_heading=h.3dy6vkm)

[3.1 Các phương pháp học máy được sử dụng. 6](#_heading=h.1t3h5sf)

[3.1.1 Naive bayes 6](#_heading=h.2s8eyo1)

[3.1.2 Logistic Regression 7](#_heading=h.3rdcrjn)

[3.1.3 Random Forest 7](#_heading=h.26in1rg)

[3.1.4 Deeplearning neural network 9](#_heading=h.35nkun2)

[3.2 Huấn luyện và lựa chọn mô hình. 10](#_heading=h.44sinio)

[Chương 4: Kết quả chạy thực nghiệm 12](#_heading=h.z337ya)

[Tài liệu tham khảo 13](#_heading=h.1y810tw)

# Chương 1: Phát biểu bài toán

Phân loại văn bản hay còn gọi là Text Classifcation hoặc là Text Categorizer là một bài toán thuộc về lĩnh vực xử lý ngôn ngữ tự nhiên dưới dạng văn bản (text). Tuy nhiên nó gắn liền với machine learning do có liên quan đến phân loại từ phân loại

Bài toán phân loại văn bản là một bài toán rất hay cũng như khá phức tạp. Bài toán đặt ra nếu như ta có một văn bản thuộc một thể loại nào đó, thì mong muốn, tương ứng với văn bản đó thì thể loại tương ứng của văn bản đó sẽ là gì. Nếu như có thể dự đoán được thể loại văn bản đó thì việc nhóm các văn bản nãy sẽ dễ dàng hơn. Từ đó đưa ra các đề xuất tương ứng. Bài toán khá phức tạp là có rất nhiều thể loại văn bản với độ dài ngắn khác nhau. Thêm nữa là sự phức tạp của ngôn ngữ tự nhiên.

Với bài toán phân loại văn bản này, dữ liệu huấn luyện bao gồm 33759 samples và testing là 50373 samples. Chia đều cho các thể loại văn bản. Các phương pháp để giải quyết sẽ được trình bày chi tiết ở các mục bên dưới.

# 

# 

# Chương 2: Tiền xử lý dữ liệu

## 2.1 Các bước tiến hành.

Đối với hầu hết các giải thuật machine learning, bước tiền dữ liệu đóng vai trong rất quan trọng. Thông qua tiền xử lý dữ liệu, ta muốn thu được các dữ liệu chứa các thông tin hữu ích cho bài toán của chúng ta, loại bỏ đi các không tin nhiễu (thông tin không mang lại ý nghĩa gì) và các điểm dữ liệu (tức các sample) có kích thước như nhau. Với bài toán gán nhãn văn bản bước tiền xử lý dữ liệu được thực hiện như sau:

* Chuyển đổi tất cả các kí tự về chữ viết thường, chỉnh lại các từ theo quy tắc âm tiết của tiếng việt, đưa văn bản về dạng chuẩn giữa hai chữ trong một văn bản thì được cách nhau bởi một dấu cách. Loại bỏ các kí tự đặc biệt như: ; , ./?...
* Loại bỏ các stop words.
* Thực hiện biểu diễn dạng bag of words và IF-IDF.

Với các từ nếu hai từ giống nhau nhưng được viết khác nhau như “Tôi” và “tôi” nếu không chuyển về cùng một từ thì ở bước biểu diễn thứ ba ta sẽ tạo ra một feature dư thừa. Để thực hiện điều này ta chuyển đổi tất cả các chữ trong các văn bản về dưới dạng chữ viết hoa. Cũng từ mục đích chỉ giữa lại các đặc trưng hữu ích. Dựa vào quy tắc âm tiết kề nhau thì phải khác nhau.

Trong dữ liệu văn bản có rất nhiều ký tự viết tắt. Nếu không thực hiện chuyển các ký tự này về dạng chữ gốc của nó. Sau khi thực hiện bag of words hay IF-IDF sẽ tạo ra ra nhiều chiều dữ liệu dư thừa. Để thực hiện chuyển đổi này cách được xây dựng một từ điển lưu trữ các từ viết tắt và từ đầy đủ của nó. Từ điển này được xây dựng dựa trên tập huấn luyện 33759 mẫu.

Loại bỏ các stop words. Các stop words chứa các thông tin dư thừa nên loại bỏ chúng sẽ làm cho dữ liệu tinh hơn. Ví dụ như từ “và”, “hoặc”, … trong bài toán phân loại không mang ý nghĩa gì. Danh sách các stop words được sưu tầm từ trên mạng.

\*Biểu diễn dạng bag of words: dữ liệu sau khi qua các bước ở trên sẽ được biểu diễn dưới dạng để máy tính có thể hiểu được. Các bước thực hiện:

* Xây dựng copus từ các văn bản. Kích thước của copus (tức số lượng các từ) sẽ là số chiều của một vector dùng để biểu diễn văn bản. Trong quá trình testing ta chỉ quan tâm đến các từ có trong copus, các từ không có trong copus sẽ không được xem xét.
* Với mỗi văn bản sẽ được biểu diễn dưới dạng một vector v với thể hiện số lượng lần xuất hiện của từ thứ trong văn bản đó.

\*Biểu diễn dữ liệu dạng TF-IDF: là một phương pháp thống kê nhằm xác định độ quan trọng của một từ trong đoạn văn bản hoặc trong một tập văn bản khác nhau. Thì gồm có hai thành phần.

Thành phần TF là tuần số xuất hiện của một từ trong một văn bản , là tần số xuất hiện của từ t trong văn bản d, T là tổng số từ trong văn bản đó.

Thành phần IDF: tính toán độ quan trọng của một từ. khi tính TF, mỗi từ đều có mối quan trọng như nhau, nhưng một số từ sẽ không thực sử quan trọng. Khi đó idf(t) = trong đó N là tổng số các văn bản và tập | là số văn bản chứa từ t.

Khi đó tf-idf(t) = tf(t) \* idf(t)

## 2.2 Ưu điểm và nhược điểm của cách tiền xử lý dữ liệu

Với cách tiền xử lý dữ liệu như trên có một số ưu điểm và nhược điểm như sau:

|  |  |
| --- | --- |
| Ưu điểm: | Nhược Điểm |
| + Loại bỏ được các thông tin không cần thiết. Đưa dữ liệu về dạng ngắn gọn nhưng vẫn không làm mất mát đi các thông tin hữu ích cho bài toán phần lại (ba bước đầu tiên).  + Với các biểu diễn bag of words chuyển đổi các văn bản về một vector có độ dài cố định. Máy tính có thể hiểu được  +Với biểu diễn dạng TF-IDF thể hiện được sự quan trọng của các từ | + Bag of words chỉ đơn giản thể hiện được tần số xuất hiện của các từ trong một văn bản. Chưa thể hiện được mức độ quan trọng của các từ.  + Bag of words và TF-IDF chưa thể hiện được thứ tự xuất hiện được các chữ trong một văn bản. Với các trường hợp khi ta đảo chỗ vị trí các từ thì dẫn đến ngữ nghĩa bị thay đổi thì chính là điểm yếu của hai cách biểu diễn trên. |

# 

# Chương 3: Lựa chọn mô hình

## 3.1 Các phương pháp học máy được sử dụng.

Với lượng dữ liệu vừa đủ lớn có thể sử dụng các giải thuật machine learning truyền thống hoặc các giải thuật deep learning. Với bài toán phân loại văn bản này, các giải thuật học máy và deeplearning được sử dụng là: Bernoulli Naive Bayes, Logistic Regression, Deep Neural Network, Random Forest. Các mô hình này sẽ được mô ta chi tiết trong phần tương ứng với các giải thuật đó ở bên dưới.

### 

### **3.1.1 Bernoulli Naive Bayes**

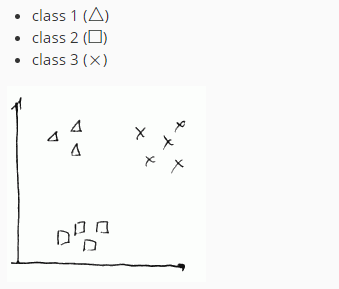
Mô hình này được áp dụng cho các loại dữ liệu mà mỗi thành phần là một giá trị nhị phân bằng 0 hoặc 1. Ví dụ, cùng với dữ liệu văn bản nhưng thay vì đếm tổng số lần xuất hiện của một tử trong văn bản, ta chỉ quan tâm đến từ đó có xuất hiện hay không. Việc biểu diễn này sẽ làm mất thông tin vốn có ban đầu của dữ liệu. Khi đó

Với có thể được hiểu là xác suất từ thứ xuất hiện trong các văn bản của lớp c.

### **3.1.2 Logistic Regression**

Logistic Regression là một bài toán dùng để phân loại hai lớp, vậy để có thể ứng dụng vào bài toán phân loại văn bản với 10 lớp này ta cần sử dụng one vs all.

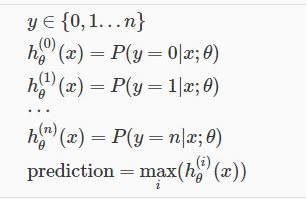
Tư tưởng chính của phương pháp này là vẫn phải qui về dạng nhãn 0 và 1. Giả sử chúng ta có 3 class như sau:



Ta phải xây dựng phải 3 hàm dự đoán như sau:

* Hàm thứ nhất dự đoán thuộc class thứ nhất (△) hoặc không thuộc.
* Hàm thứ hai dự đoán thuộc class thứ hai (□) hoặc không thuộc.
* Hàm thứ ba dự đoán thuộc class thứ ba (×) hoặc không thuộc.

Với một biến đầu vào ta sử dụng cả ba hàm để lấy kết quả. Hàm nào của nhãn có kết quả cáo nhất thì dự đoán chính là output theo công thức sau



### **3.1.3 Random Forest**

Random Forest-Rừng ngẫu nhiên là một thuật toán học có giám sát. Như tên gọi của nó, Rừng ngẫu nhiên sử dụng các cây (tree) để làm nền tảng.

Rừng ngẫu nhiên là một tập hợp của các Decision Tree, mà mỗi cây được chọn theo một thuật toán dựa vào ngẫu nhiên.

Decision Tree là tên đại diện cho một nhóm thuật toán phát triển dựa trên cây quyết định. Ở đó, mỗi Node của cây sẽ là các thuộc tính, và các nhánh là giá trị lựa chọn của thuộc tính đó. Bằng cách đi theo các giá trị thuộc tính trên cây, cây quyết định sẽ cho ta biết giá trị dự đoán.

Điểm mạnh của thuật toán:

* Random Forest algorithm có thể sử dụng cho cả bài toán Classification và Regression.
* Random Forest làm việc được với dữ liệu thiếu giá trị.
* Khi Forest có nhiều cây hơn, chúng ta có thể tránh được việc Overfitting với tập dữ liệu.
* Có thể tạo mô hình cho các giá trị phân loại.

Random Forest hoạt động bằng cách đánh giá nhiều cây quyết định ngẫu nhiên, và lấy ra kết quả được đánh giá tốt nhất trong số kết quả trả về.

Mã giả cho hoạt động của Random Forest:

* + - 1. Chọn ngẫu nhiên k features từ tập m features (k<<m).
      2. Từ tập k features, tính toán ra node d là tốt nhất cho Node phân loại.
      3. Chia các node con theo node tốt nhất vừa tìm được.
      4. Lặp lại bước 1-3 cho đến khi đạt đến k node.
      5. Lặp lại bước 1-4 để tạo ra n cây*.*

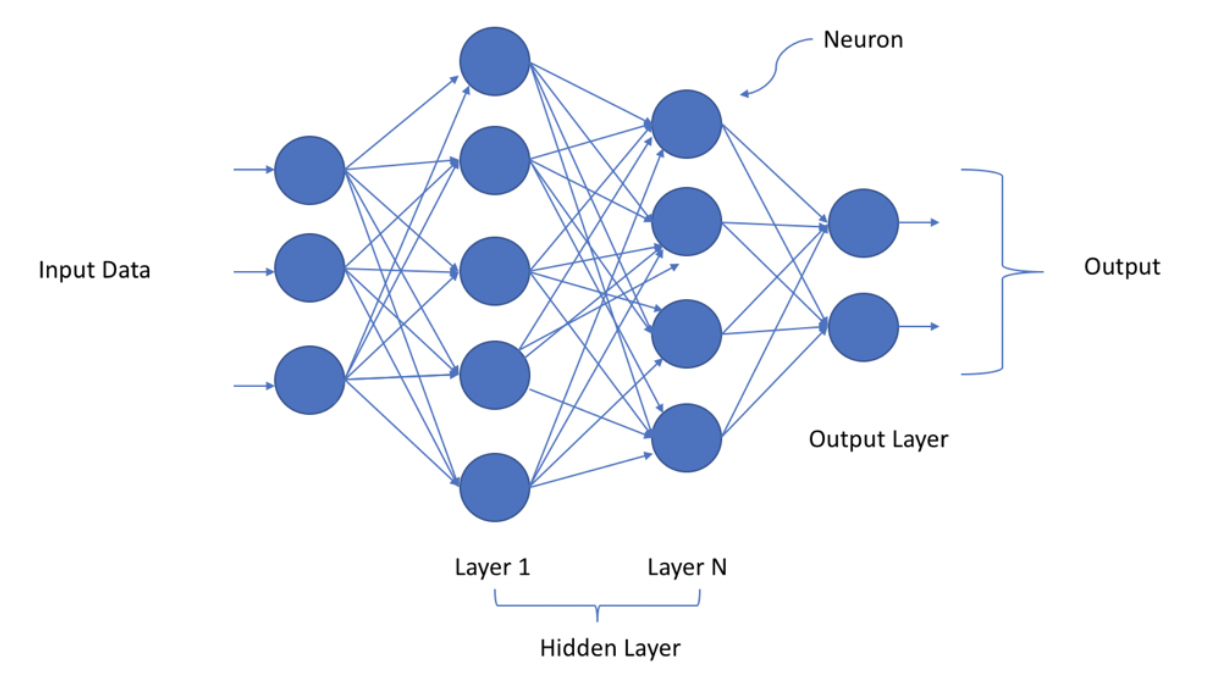
Random Forest Prediction:

1. Lấy các test features và sử dụng các Cây quyết định đã tạo ra để dự đoán kết quả, lưu nó vào một danh sách.
2. Tính toán số lượng vote trên toàn bộ Forest cho từng kết quả.
3. Lấy kết quả có số lượng vote lớn nhất làm kết quả cuối cho mô hình.

### **3.1.4 Deeplearning neural network**

## Neural là tính từ của neuron (nơ-ron), network chỉ cấu trúc đồ thị nên neural network (NN) là một hệ thống tính toán lấy cảm hứng từ sự hoạt động của các nơ-ron trong hệ thần kinh. Neural network chỉ là lấy cảm hứng từ não bộ và cách nó hoạt động, chứ không phải bắt chước toàn bộ các chức năng của nó. Việc chính của chúng ta là dùng mô hình đấy đi giải quyết các bài toán chúng ta cần.

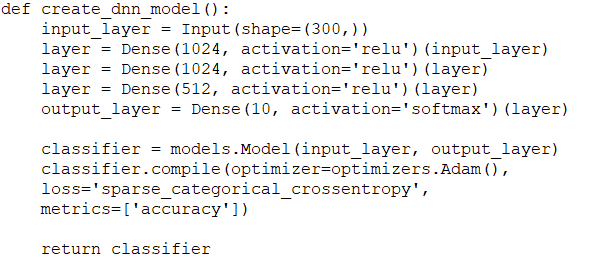
Mô hình neural network cơ bản:



Layer đầu tiên là input layer, các layer ở giữa được gọi là hidden layer, layer cuối cùng được gọi là output layer. Các hình tròn được gọi là node.

Mỗi mô hình luôn có 1 input layer, 1 output layer, có thể có hoặc không các hidden layer. Tổng số layer trong mô hình được quy ước là số layer -1.

Trong bài toán này sử dụng, 1 input layer, 3 hidden layer và 1 ouput layer được mô tả chi tiết như sau.



## 3.2 Huấn luyện và lựa chọn mô hình.

Trước tiên để đánh giá một mô hình là tốt hay xấu ta dùng độ đo là accuracy. Với tập dữ liệu bao gồm 33759 tập mẫu để huấn luyện ta tiến hành chia tập dữ liệu thành hai tập riêng việc là tập train và tập valid. Khi đó nếu tập valid chia quá ít sample thì kết quả đánh giá trên nó chưa chắc phản ảnh đúng chất lượng của mô hình học máy. Còn khi cho tập valid lớn, khi đó số lượng tập train sẽ ít đi điều này sẽ ảnh hưởng trực tiếp chất lượng của mô hình. Lúc này mô hình rất dễ chỉ được huấn luyện trên một vùng nhất định trong không gian của dữ liệu chứ nó chưa đặc tả hết toàn bộ không gian của dữ liệu.

Với lượng dữ liệu lớn ta có thể tiến hành chia tập dữ liệu train ra làm 2 phần là: tập train mới, tập valid theo tỉ lệ 9:1 tức là 90 % dữ liệu được dung cho việc training và 10 % dung cho việc test. Sau khi training qua các mô hình: Bernoulli Naive Bayes, Logistic Regression, Deep Neural Network, Random Forest ta có kết quả sẽ được trình bày trong phần dưới đây.

# Chương 4: Kết quả chạy thực nghiệm

Chạy thực nghiệm trên bộ dữ liệu với 33759 tập mẫu. Các thí nghiệm được cài đặt các thí nghiệm sau:

* Exp1. Naive Bayes.
* Exp2. Logistic Regression.
* Exp3. Random Forest.
* Exp4. Deeplearning Neural Network.

Kết quả được thể hiện trong bảng dưới đây:

|  |  |  |
| --- | --- | --- |
| Experiment | Valid Accuracy | Validation Accuracy |
| Exp1 | 78.99% | 78.89% |
| Exp2 | 87.32% | 86.42% |
| Exp3 | 85.46% | 84.25% |
| Exp4 | **88.30%** | **87.57%** |

Kết luận: Sau khi thực hiện các thí nghiệm, ta thấy phương pháp Naive bayes không cho kết quả tốt. Nó xuất pháp từ chính nhược điểm của phương pháp Naive bayes (xác suất hậu nghiệm của các nhãn trong mô hình). Mô hình Deeplearning cho kết quả tốt nhất.

# Tài liệu tham khảo

Machine Learning cơ bản book

Pattern recognition and machine learning