BOR3

TRƯỜNG ĐẠI HỌC KHOA HỌC – TỰ NHIÊN

KHOA TOÁN - TIN HỌC



**BÁO CÁO TIỂU LUẬN**

**PHƯƠNG PHÁP TOÁN TRONG TIN HỌC**

**ĐỀ TÀI: HỆ THỐNG ĐỀ XUẤT CẦU THỦ BÓNG ĐÁ.**

SINH VIÊN : BIỆN HUỲNH HỮU THỊNH

MÃ SỐ SINH VIÊN : 1611260

KHOA : TOÁN – TIN HỌC

CHUYÊN NGHÀNH : PHƯƠNG PHÁP TOÁN TRONG TIN HỌC

GIẢNG VIÊN HƯỚNG DẪN: TS. TRẦN ANH TUẤN

TP. HCM, 04/2019 – 12/2019

# LỜI CẢM ƠN

Tôi xin chân thành cảm ơn Khoa Toán - Tin học trường đại học Khoa Học Tự Nhiên đã tạo điều kiện thuận lợi cho tôi học tập và thực hiện đề tài tốt nghiệp này.

Tôi xin bày tỏ lòng biết ơn sâu sắc tới thầy Trần Anh Tuấn đã tận tình hướng dẫn chỉ bảo tôi trong quá trình thực hiện đề tài.

Tôi xin chân thành cảm ơn quý thầy cô trong khoa đã tận tình giảng dạy, trang bị cho tôi những kiến thức quý báu trong năm vừa qua.

Tôi xin chân thành cảm ơn ông bà, cha mẹ đã luôn động viên ủng hộ vật chất lẫn tinh thần trong suốt thời gian qua.

Tôi xin cảm ơn sự quan tâm giúp đỡ và ủng hộ của các anh chị bạn bè trong quá trình thực hiện khóa đề tài. Mặc dù đã cố gắng hoàn thành khóa luận trong phạm vi và khả năng cho phép nhưng chắc chắn sẽ không tránh khỏi những thiếu sót.

Tôi rất mong nhận được sự thông cảm, góp ý và tận tình chỉ bảo của quý thầy cô và các bạn.

TP. Hồ Chí Minh, tháng 1 năm 2020

Sinh viên thực hiện

Biện Huỳnh Hữu Thịnh

# GIỚI THIỆU

Cùng với sự phát triển của thương mại điện tử cũng như tương tác trực tuyến của con người trong giai đoạn hiện nay, hệ thống gợi ý đã được nghiên cứu và ứng dụng một cách mạnh mẽ và mang lại lợi ích cho cả người cung cấp dịch vụ và người sử dụng dịch vụ. Ý tưởng đằng sau một hệ thống gợi ý là sử dụng các dữ liệu thu thập được từ người dùng và dự đoán, gợi ý cho người dùng những sản phẩm, tính năng, dịch vụ mà người dùng có thể thích, từ đó nâng cao được chất lượng dịch vụ và thu lại lợi nhuận.

Ví dụ như:

* Youtube tự động chuyển các clip liên quan đến clip mình đang xem. Youtube cũng tự gợi ý những clip mà có thể mình sẽ thích.
* Khi mình mua một món hàng trên Amazon, hệ thống sẽ tự động gợi ý “Frequently bought together”, hoặc nó biết mình có thể thích món hàng nào dựa trên lịch sử mua hàng của mình.
* Facebook hiển thị quảng cáo những sản phẩm có liên quan đến từ khoá mình vừa tìm kiếm.
* Facebook gợi ý kết bạn.
* Netflix tự động gợi ý phim cho người dùng.

Hệ thống đề xuất ngoài tạo sự tiện lợi, thoải mái cho người dùng còn có thể làm tăng tỷ lệ mua hàng của khách hàng cũng như thời gian giữ khách ở lại với website của mình.

**MỤC LỤC**

[LỜI CẢM ƠN 2](#_Toc29237798)

[GIỚI THIỆU 3](#_Toc29237799)

[I. BÀI TOÁN 5](#_Toc29237800)

[II. CÁC HƯỚNG TIẾP CẬN 6](#_Toc29237801)

[III. HƯỚNG TIẾP CẬN DỰA TRÊN ĐẶC TÍNH 6](#_Toc29237802)

[III.1. Ưu điểm 7](#_Toc29237803)

[III.2. Khuyết điểm 7](#_Toc29237804)

[IV. HƯỚNG TIẾP CẬN DỰA THEO LỌC CỘNG TÁC 8](#_Toc29237805)

[IV.1. Similarity function 8](#_Toc29237806)

[IV.2. Dự đoán điểm đánh giá. 9](#_Toc29237807)

[V. CÁC PHƯƠNG PHÁP MÁY HỌC 10](#_Toc29237808)

[VI. ĐÁNH GIÁ HỆ THỐNG ĐỀ XUẤT 10](#_Toc29237809)

[VI.1. Đánh giá dựa trên các xếp hạng. 10](#_Toc29237810)

[VI.2. Đánh giá dựa trên các đề xuất 10](#_Toc29237811)

[VII. TÀI LIỆU THAM KHẢO 12](#_Toc29237812)

# BÀI TOÁN

Bài toán là một chức năng trong phần mềm mà tôi đang thiết kế. Trong bước chọn đội hình của mình, nhiều người dùng thường gặp khó khăn trong việc chọn các cầu thủ đã định hình trước của mình.

Trong bóng đá, một đội bóng có thể có nhiều đội hình trong suốt một trận đấu nhưng thường họ chỉ chọn một đội hình cụ thể. Và một cơ sở dữ liệu về cầu thủ chứa rất nhiều thông tin về nhiều cầu thủ khác nhau qua từng năm. Người dùng rất thường mất nhiều thời gian tìm kiếm chính xác - như tìm theo tên, câu lạc bộ, quốc tịch. Vì vậy tôi mong muốn phần mềm của mình có thể hỗ trợ một phần cho người dùng, bằng cách đề xuất một danh sách các cầu thủ có thể họ đang cần thêm, qua đó giúp cho thao tác người dùng nhanh gọn và thuận tiện hơn.

Danh sách thông tin thuộc tính của cầu thủ được cung cấp bởi rất nhiều nguồn, điển hình nhất là FIFA Index [5] hoặc FIFA addict [6]. Dữ liệu về cầu thủ mà tôi dùng trong bài tiểu luận này được khai thác từ trang FIFA addict [6], gồm thông tin của hơn 20,000 cầu thủ. Về phần lịch sử người dùng do phần mềm tôi thu thập khoảng 90 người dùng và 1,000 dòng dữ liệu.

Do thời gian chuẩn bị cho bài tiểu luận này có hạn nên tôi chỉ tập trung nghiên cứu về các hệ thống đề xuất cơ bản và giản đơn.

# CÁC HƯỚNG TIẾP CẬN

Trong bài toán này đối tượng mà hệ thống gợi ý hướng đến được gọi là người dùng, còn sản phẩm mà hệ thống đưa ra gợi ý gọi là cầu thủ. Hệ thống thường dựa trên tương tác trong quá khứ giữa người dùng và cầu thủ để tạo nên dự đoán trong tương lai.

Các Hệ thống đề xuất thường được chia thành hai nhóm lớn [4]:

* Hướng tiếp cận dựa trên đặc tính (Content-based systems).
* Hướng tiếp cận dựa theo lọc cộng tác (Collaborative filtering).

Mỗi hướng tiếp cận đều có điểm mạnh và hạn chế riêng của chúng. Tôi sẽ trình bày chi tiết ở phần tiếp theo.

Như đã đề cập, có hai thực thể chính trong các hệ thống đề xuất là người dùng và cầu thủ. Mỗi người dùng sẽ có mức độ quan tâm (degree of preference) tới từng cầu thủ khác nhau. Mức độ quan tâm này, nếu đã biết trước, được gán cho một giá trị ứng với mỗi cặp người dùng – cầu thủ ta tạm gọi giá trị này là điểm đánh giá. Tập hợp tất cả các điểm đánh giá, bao gồm cả những giá trị chưa biết cần được dự đoán, tạo nên một ma trận gọi là utility matrix.

Trong bài toán này các giá trị trong utility matrix là giá trị nhị phân với giá trị 1 là cầu thủ đó có trong đội hình và 0 là cầu thủ chưa có trong đội.

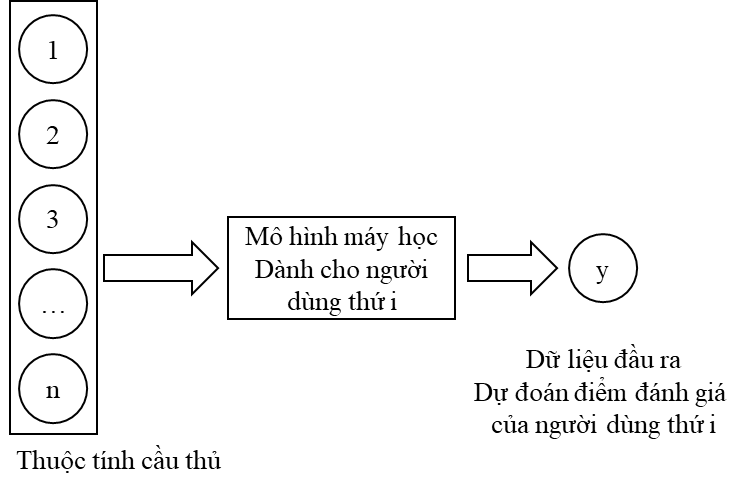
# HƯỚNG TIẾP CẬN DỰA TRÊN ĐẶC TÍNH

Hệ thống đề xuất cầu thủ dựa trên đặc tính, chỉ số của họ. Ví dụ một người dùng sử dụng rất nhiều cầu thủ có tốc độ cao thì có thể họ cũng dùng các vị trí khác cũng là cầu thủ có tốc độ cao.

Cách tiếp cận này yêu cầu chúng ta cần xây dựng một bộ hồ sơ - danh sách các thuộc tính - cho mỗi cầu thủ cầu thủ được biểu diễn dưới dạng một vectơ (feature vector).

Ứng với mỗi người dùng ta tìm một mô hình máy học để có thể dự đoán xem họ có cầu thủ đó trong đội hình của mình hay không. Bài toán đi tìm mô hình cho mỗi người dùng có thể được coi là một bài toán hồi quy (regression) trong trường hợp điểm đánh giá là một đoạn giá trị, hoặc bài toán phân lớp (classification) trong trường hợp điểm đánh giá là một vài trường hợp cụ thể, như thích hoặc không thích chẳng hạn.

Dữ liệu huấn luyện bao gồm nhiều cặp “hồ sơ cầu thủ - nhãn đánh dấu cầu thủ đó có trong đội hình của người dùng”.



Hình 3.1: Mô hình máy học áp dụng cho từng người dùng.

## Ưu điểm

Sử dụng hết dữ liệu sẵn có bao gồm hồ sơ của cầu thủ, và danh sách đội hình đã thu thập được từ lịch sử người dùng. Có thể áp dụng nhiều mô hình máy học, mô hình học sâu (deep learning model) - ví dụ như: Hồi quy tuyến tính (Linear regression), Phân nhóm Naive Bayes (Naive Bayes Classifier), các loại mạng nơron … - cho từng trường hợp cụ thể.

## Khuyết điểm

Xây dựng cho mỗi người dùng một mô hình máy học riêng biệt hao tốn nhiều dung lượng của hệ thống. Một mô hình thường sử dụng nhiều trọng số, một hệ thống có nhiều người dùng, vì vậy khi người dùng càng nhiều thì ta cần một lượng lớn không gian để lưu trữ mô hình này.

Thời gian huấn luyện một mô hình máy học không ít, nên như trường hợp trên việc huấn luyện từng mô hình cho mỗi người dùng cũng mất một lượng lớn thời gian.

Như bạn đã biết việc đáp ứng dữ liệu huấn luyện cho một mô hình cần rất nhiều dữ liệu mẫu. Cụ thể trong trường hợp này, mỗi người dùng thường chỉ thêm khoảng 10-40 cầu thủ vào đội hình của mình. Nên với bộ dữ liệu huấn luyện chỉ có 10-40 trường hợp để huấn luyện cho mô hình dự đoán trên tập khoảng 20,000 cầu thủ thì độ chính xác không có được sự tin tưởng cao.

Khi xây dựng mô hình cho một người dùng, hệ thống không tận dụng được thông tin từ các người dùng khác.

# HƯỚNG TIẾP CẬN DỰA THEO LỌC CỘNG TÁC

Ý tưởng cơ bản của phương pháp này là xác định mức độ quan tâm của một người dùng tới một cầu thủ dựa trên những người dùng khác có điểm gần giống với họ. Hoặc ngược lại, xác định mức độ quan tâm của người dùng tới một cầu thủ dựa trên những cầu thủ khác có thuộc tính tương đồng. Dữ liệu duy nhất chúng ta có là utility matrix.

Ví dụ 1: A, B đều thích các cầu thủ của câu lạc bộ Manchester United, tức họ đều sở hữu các cầu thủ của đội bóng này trong đội của mình. Ta đã biết A cũng thích Wayne Rooney (một cầu thủ đã từng thi đấu cho Manchester United), vậy nhiều khả năng B cũng có cầu thủ này trong đội.

Ví dụ 2: Sergio Ramos và Gerard Piqué thường được chọn thi đấu chung với nhau, tức là đa số người dùng đều thêm 2 cầu thủ này vào đội hình của họ. Nếu A đã có Sergio Ramos trong đội của mình thì khả năng cao A cũng có Gerard Piqué trong đội.

Việc quan trọng nhất phải làm trước tiên trong hướng tiếp cận này là phải xác định được sự giống nhau (similarity) giữa hai người dùng.

## Similarity function

Lựa chọn độ đo để tính độ tương đồng giữa hai người dùng là một bước quan trọng trong việc thiết kế hệ thống. Bởi vì, nó ảnh hưởng trực tiếp đến kết quả tư vấn của hệ thống. Hiện tại, trong lĩnh vực nghiên cứu máy học, có nhiều độ đo được đề xuất cho mục đích này. Trong đó, Pearson correlation, Cosine similarity là hai độ đo được sử dụng rộng rãi.

**Pearson correlation** là độ đo tính sự tương đồng giữa hai người dùng dựa trên tương quan thống kê [1]. Độ tương đồng của hai người dùng *u* và *v* được xác định bằng công thức (1):

|  |  |  |
| --- | --- | --- |
|  |  | ( 1) |
|  |  |  |
|  |  |  |

Với là giá trị tương đồng giữa người dùng u và người dùng v;

n là số chiều của vector (số cầu thủ);

là điểm đánh giá của người dùng u cho cầu thủ i;

là điểm đánh giá của người dùng v cho cầu thủ i;

là điểm đánh giá trung bình của người dùng u;

là điểm đánh giá trung bình dùng v;

**Cosine similarity** là độ đo tính sự tương đồng giữa hai người dùng dựa trên không gian vector đại số tuyến tính. Các điểm đánh giá của từng người dùng trên n cầu thủ hay mục dữ liệu được biểu diễn bằng một vector n chiều. Độ tương đồng của hai người dùng u và v được xác định bằng khoảng cách Cosine giữa hai vector và vector theo công thức (2):

|  |  |  |
| --- | --- | --- |
|  |  | ( 2) |
|  |  |  |

Với là giá trị tương đồng giữa người dùng u và người dùng v;

n là số chiều của vector (số cầu thủ);

là điểm đánh giá của người dùng u cho cầu thủ i;

là điểm đánh giá của người dùng v cho cầu thủ i;

## Dự đoán điểm đánh giá.

Để tính toán kết quả khuyến nghị cho người dùng u, bước đầu tiên, hệ tư vấn lọc cộng tác dựa trên người dùng u sử dụng các độ đo tương đồng để tìm ra danh sách k người dùng tương đồng với người dùng u. Khi có danh sách k người dùng tương đồng, hệ thống sẽ kết hợp các giá trị xếp hạng của họ để sinh ra dự đoán sở thích của người dùng u đối với cầu thủ i. Thông thường, kết quả dự đoán được tính dựa trên trọng số trung bình của giá trị xếp hạng của k người dùng tương đồng được biểu diễn bằng công thức (3).

|  |  |  |
| --- | --- | --- |
|  |  | ( 3) |

Với là điểm đánh giá dự tính nếu người dùng u đánh giá cho cầu thủ i;

U là tập hợp k người dùng có giá trị tương đồng gần nhất bới người dùng u;

là điểm đánh giá của người dùng cho cầu thủ i;

là điểm đánh giá của người dùng u cho cầu thủ i;

# CÁC PHƯƠNG PHÁP MÁY HỌC

Ở phần này, tôi sẽ giới thiệu tổng quan về các phương pháp máy học có thể sử dụng trong phương pháp đề xuất dự trên đặc tính.

## Hồi quy tuyến tính

Đây là một thuật toán học có giám sát - Supervised learning. Bài toán này đôi khi được gọi là Linear Fitting hoặc Linear Least Square. Phương pháp này dùng một hàm tuyến tính để xấp xỉ kết quả thực, sau đó từ hàm tuyến tính đó dự đoán các kết quả chưa có.

|  |  |  |
| --- | --- | --- |
|  |  | ( 4) |

Trong phương trình (4) phía trên, nếu chúng ta đặt là vector hệ số cần phải tối ưu và là vector (hàng) dữ liệu đầu vào mở rộng. Số “1” ở cuối được thêm vào để phép tính đơn giản hơn và thuận tiện cho việc tính toán. Khi đó, phương trình (4) có thể được viết lại dưới dạng:

|  |  |  |
| --- | --- | --- |
|  |  | ( 5) |

Chúng ta mong muốn rằng sự sai khác giữa giá trị thực y và giá trị dự đoán là nhỏ nhất. Nói cách khác, làm hàm mất mát của Y và càng nhỏ càng tốt. Có nhiều dạng của hàm mất mát, trong phương pháp này ta sử dụng hàm mất mát (6) để dễ tính toán đạo hàm.

|  |  |  |
| --- | --- | --- |
|  | với n là số chiều của y | ( 6) |
|  |  |  |

Để tìm được giá trị cực tiểu toàn cục của hàm nhiều chiều thường rất phức tạp và tốn một khối lượng lớn phép toán để tính ra, thậm chí có một số hàm không có cực tiểu. Thay vào đó, người ta thường cố gắng tìm các điểm cực tiểu địa phương - local minimum, và ở một mức độ nào đó, coi đó là nghiệm cần tìm của bài toán. Ta có một số phương pháp để tìm cực tiểu địa phương như phương pháp đạo hàm giảm dần (Gradient descent), Adagrad, Adadelta, RMSprop… [9]. Tôi sẽ sử dụng phương pháp gradient descent để tối ưu trọng số (w), ta sẽ lặp lại phương trình (7) đến khi hàm J(W) hội tụ.

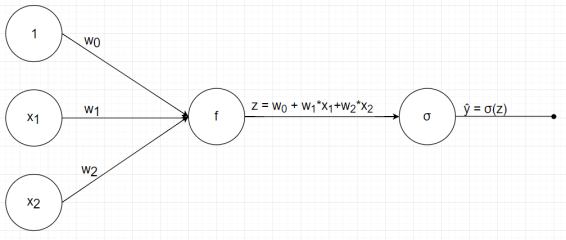
|  |  |  |
| --- | --- | --- |
|  | với hệ số học máy | ( 7) |

Khi hàm J(W) đã hội tụ, ta ghi nhớ trọng số w để sử dụng lại trong quá trình dự đoán các yêu cầu trong tương lai.

## Hồi quy logistic

Đây là một phương pháp học có giám sát, dùng để dự đoán các kết quả có hai giá trị (Binary classification). Thay vì đưa ra kết luận phân lớp, thì phương pháp đưa xác suất khả năng thực thể (object) cần xét thuộc vào từng phân lớp.

Phương pháp sẽ sử dụng một hàm để biến đổi hàm tuyến tính (trong hồi quy tuyến tính) thành phi tuyến, được gọi là hàm kích hoạt (activate function). Các hàm kích hoạt của hồi quy logistic phải thỏa mãn các yêu cầu sau: hàm số liên tục, bị chặn trong khoảng (0, 1); hàm trơn (smooth) để tính đạo hàm ở mọi điểm.



Hình 5.2.1: Sơ đồ mô tả phương pháp hồi quy logistic.

Hồi quy logistic được viết dưới dạng chung dưới đây:

|  |  |  |
| --- | --- | --- |
|  | với là hàm kích hoạt | ( 8) |

Các hàm kích hoạt thông dụng:

|  |  |  |
| --- | --- | --- |
|  |  | ( 9) |
|  |  | ( 10) |
|  |  | ( 11) |
|  |  | ( 12) |

Ta có thể giả sử rằng xác suất để một điểm dữ liệu x rơi vào class 1 là và rơi vào class 0 là . Với mô hình được giả sử như vậy, với các điểm dữ liệu training (đã biết đầu ra y), ta có thể viết như sau:

|  |  |  |
| --- | --- | --- |
|  |  | ( 13) |
|  |  | ( 14) |

Ký hiệu và viết gộp lại hai biểu thức bên trên ta có:

|  |  |  |
| --- | --- | --- |
|  |  | ( 15) |

Xét toàn bộ tập dữ liệu huấn luyện với và, chúng ta cần tìm w để biểu thức sau đây đạt giá trị lớn nhất. Giả sử thêm rằng các điểm dữ liệu được sinh ra một cách ngẫu nhiên độc lập với nhau, ta có thể viết:

|  |  |  |
| --- | --- | --- |
|  |  | ( 16) |

Việc tối ưu hàm (16) khi N lớn, tích của N số nhỏ hơn 1số nhỏ hơn 1 có thể dẫn tới sai số trong tính toán (numerial error) vì tích là một số quá nhỏ. Một phương pháp thường được sử dụng đó là lấy logarit tự nhiên của biến phép nhân thành phép cộng và để tránh việc số quá nhỏ. Sau đó lấy ngược dấu để được một hàm (17) và coi nó là hàm mất mát. Lúc này bài toán tìm cực đại, trở thành bài toán tìm cực tiểu của hàm mất.

|  |  |  |
| --- | --- | --- |
|  |  | ( 17) |
|  |  |  |

Trong máy học, logarit thập phân ít được dùng, vì vậy thường được dùng để ký hiệu logarit tự nhiên (cơ số e). Tương tự với phương pháp hồi quy tuyến tính ta cũng dùng phương pháp gradient descent để tìm được cực tiểu địa phương cho hàm mất mát.

|  |  |  |
| --- | --- | --- |
|  |  | ( 18) |
|  |  |  |

Giả sử chọn hàm kích hoạt cho mô hình hồi quy logistic là hàm sigmoid  
, gọi.

|  |  |  |
| --- | --- | --- |
|  |  | ( 19) |

Vậy thế (19) vào (18) ta có:

|  |  |  |
| --- | --- | --- |
|  |  | (20) |

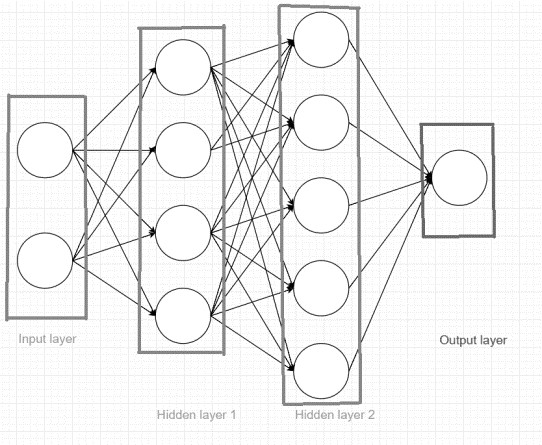
Ta sẽ lặp lại phương trình (21) để cập nhật trọng số W đến khi hàm J(W) hội tụ.

|  |  |  |
| --- | --- | --- |
|  | với hệ số học máy | ( 21) |

Khi hàm J(W) đã hội tụ, ta ghi nhớ trọng số w để sử dụng lại trong quá trình dự đoán các yêu cầu trong tương lai.

## Mạng neural nhân tạo

Có thể nói mô hình Hồi quy logistic là mô hình mạng neural đơn giản nhất chỉ với một lớp đầu vào và một lớp đầu ra.



Hình 5.3.1: Mô hình đơn giản của Mạng neural.

Trong mô hình mạng neural, lớp đầu tiên là input layer, các lớp ở giữa được gọi là hidden layer, lớp cuối cùng được gọi là output layer. Các đầu ra của mỗi lớp 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 (Không tính input layer).

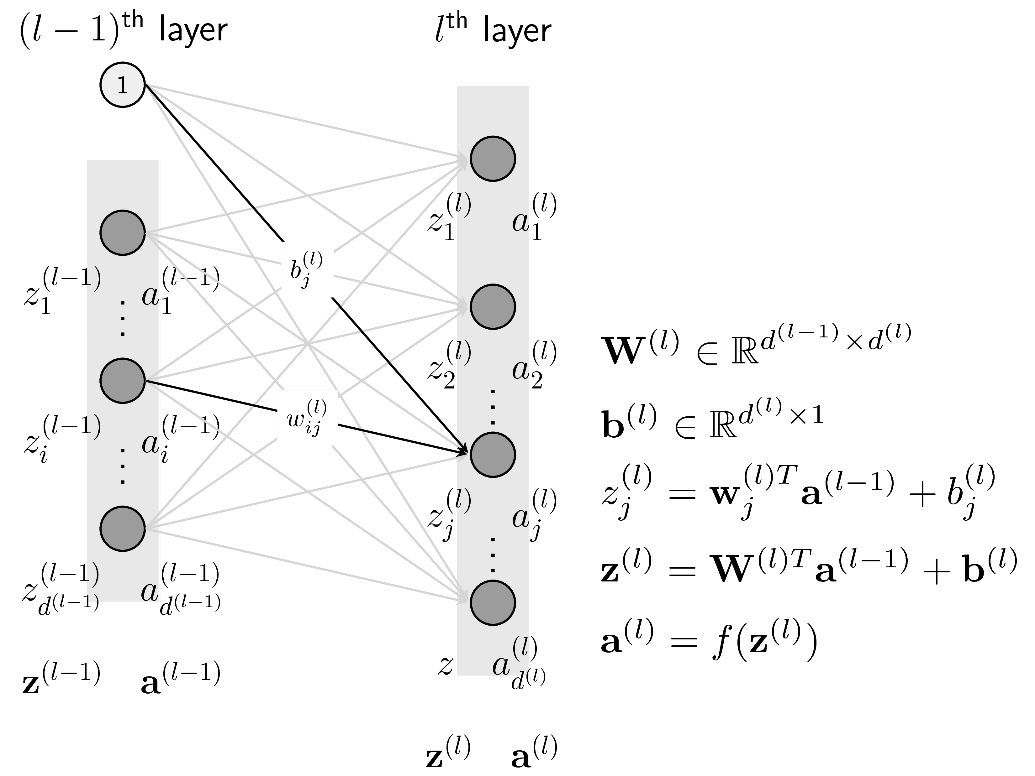
Ví dụ như ở hình trên có 1 input layer, 2 hidden layers và 1 output layer. Số lượng layer của mô hình là 3 layers.

Mỗi node trong hidden layer và output layer:

* Liên kết với tất cả các node ở layer trước đó với các hệ số w riêng.
* Mỗi node có 1 hệ số bias b riêng.
* Diễn ra 2 bước: tính tổng linear và áp dụng activation function.

Trong mỗi mô hình mạng neural có 2 quá trình chính, thứ nhất là quá trình truyền thẳng (forward propagation), và thứ 2 là quá trình lan truyền ngược (Back propagation).

Quá trình truyền thẳng khá đơn giản, chỉ cần sử dụng các node đầu ra của lớp trước đó làm dữ liệu đầu vào cho lớp tiếp theo, tiếp tục cho tới khi tới lớp cuối cùng cũng là lớp output thì ta thu được kết quả của sự lan truyền thẳng.



Hình 5.3.2: Các ký hiệu sử dụng trong mạng neural.

Quá trình lan truyền ngược cơ bản cũng dựa trên các phương pháp tối ưu hàm mất mát. Nhưng do cấu trúc mạng neural có sự liên hệ giữa các lớp, lớp sau sử dụng thông tin của lớp trước đó, nên quá trình này trở nên phức tạp hơn [10]. Như đã trình bày ở trên quá trình truyền thẳng có thể xem như các chuỗi dài các phương trình lồng nhau. Như phương trình (22):

|  |  |  |
| --- | --- | --- |
|  |  | ( 22) |

Với A, B, C là 3 hàm kích hoạt của 3 lớp khác nhau. Bằng việc áp dụng chain rule, chúng ta có thể tính được đạo hàm của theo như sau:

|  |  |  |
| --- | --- | --- |
|  |  | ( 23) |

Dựa theo cách làm trên ta có thể cập nhật trọng số ở lớp bất kì, để hàm mất mát hội tụ:

|  |  |  |
| --- | --- | --- |
|  |  | ( 24) |

Nếu là lớp output phương trình (24) có thể biến đổi như sau:

|  |  |  |
| --- | --- | --- |
|  |  | ( 25) |

Với là độ sai lệch giữa thực tế và dự đoán.

Ta làm tương tự quá trình cập nhật trọng số như hai phương pháp hồi quy trên cho tới khi hàm hội tụ.

# ĐÁNH GIÁ HỆ THỐNG ĐỀ XUẤT

Đánh giá độ chính xác của mô hình tư vấn là một khâu không thể thiếu trong quy trình xây dựng hệ thống. Nó giúp cho người thiết kế mô hình lựa chọn mô hình, kiểm tra độ chính xác của mô hình trước khi đưa mô hình vào ứng dụng thực tế. Để đánh giá mô hình tư vấn lọc cộng tác, người xây dựng hệ thống có các cách sau.

## Đánh giá dựa trên các xếp hạng.

Phương pháp này đánh giá độ chính xác của mô hình bằng cách so sánh giá trị xếp hạng dự đoán với giá trị thực hay chính xác hơn là tìm ra giá trị trung bình lỗi dựa vào ba đại lượng Root mean square error, Mean squared error và Mean absolute error. Mô hình được đánh giá là tốt khi các đại lượng này có giá trị thấp.

**Root mean square error** (RMSE): Độ lệch chuẩn giữa giá trị thực và giá trị xếp hạng.

|  |  |  |
| --- | --- | --- |
| RMSE | với | ( 26) |

**Mean squared error** (MSE): Trung bình của bình phương độ lệch giữa giá trị thực và giá trị xếp hạng dự đoán.

|  |  |  |
| --- | --- | --- |
| MSE |  | (27) |

**Mean absolute error** (MAE): Trung bình trị tuyệt đối của độ lệch giữa giá trị thực và giá trị xếp hạng dự đoán.

|  |  |  |
| --- | --- | --- |
| MSE | với | ( 28) |

Với Y là tập tất cả các điểm đánh giá thực của người dùng cho cầu thủ;

là tập tất cả các điểm đánh giá dự đoán của người dùng cho cầu thủ.

## Đánh giá dựa trên các đề xuất

Phương pháp này đánh giá độ chính xác của mô hình bằng cách so sánh các đề xuất của mô hình đưa ra với các lựa chọn thực của người dùng. Phương pháp này sử dụng ma trận hỗn độn 2x2 (Confusion matrix) để tính độ chính xác (Precision), độ bao phủ (Recall) và trung bình điều hòa giữa độ chính xác và độ bao phủ (F-measure). Mô hình được đánh giá là tốt khi ba chỉ số trên có giá trị cao.

Bảng 5.2: Ma trận hỗ độn

|  |  |  |
| --- | --- | --- |
| Lựa chọn của người dùng | Hệ thống đề xuất | |
| Có | Không |
| Có | TP | FN |
| Không | FP | TN |

Trong đó:

TP: Những cầu thủ được đề xuất đã có trong đội hình của người dùng.

FP: Những cầu thủ được đề xuất không có trong đội hình của người dùng.

FN: Những cầu thủ không đề xuất đã có trong đội hình của người dùng.

TN: Những cầu thủ không đề xuất không có trong đội hình của người dùng.

**Độ chính xác** (Precision):

|  |  |  |
| --- | --- | --- |
| Precision |  | ( 29) |

**Độ bao phủ** (Recall):

|  |  |  |
| --- | --- | --- |
| Recall |  | ( 30) |

**Trung bình điều hòa giữa độ chính xác và độ bao phủ** (F-Measure):

|  |  |  |
| --- | --- | --- |
| F-Measure |  | ( 31) |

# TÀI LIỆU THAM KHẢO

[1] Niels Mündler, “Association rule mining and itemset-correlation based variants”, [Online]. Available: <https://arxiv.org/pdf/1907.09535.pdf>.

[2] Rafael Glauber and Angelo Loula, “Collaborative Filtering vs. Content-Based Filtering: differences and similarities”, [Online]. Available: <https://arxiv.org/pdf/1912.08932.pdf>.

[3] Premier League News, “2019/20 Premier League squads confirmed”, [Online]. Available: <https://www.premierleague.com/news/1335777>.

[4] Jure Leskovec, Anand Rajaraman and Jeffrey D. Ullman, "Mining of Massive Datasets" Recommendation Systems, pp. 324, 2012.

[5] FIFA Index, “Player Stats Database - FIFA 20”, [Online]. Available: https://www.fifaindex.com.

[6] FIFA addict, “Thông tin cầu thủ Dữ liệu cầu thủ Tiềm năng FIFA Online 4”, [Online]. Available: https://vn.fifaaddict.com.

[7] Suryakant and Tripti Mahara, “A New Similarity Measure Based on Mean Measure of Divergence for Collaborative Filtering in Sparse Environment”, [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1877050916311644>.

[8] Wikipedia, the free encyclopedia, “Pearson correlation coefficient”, [Online]. Available: <https://en.wikipedia.org/wiki/Pearson_correlation_coefficient>.

[9] Sebastian Ruder, “An overview of gradient descent optimization algorithms”, [Online]. Available: https://ruder.io/optimizing-gradient-descent.

[10] ML Glossary, “Backpropagation”, [Online]. Available: https://ml-cheatsheet.readthedocs.io/en/latest/backpropagation.html#chain-rule-refresher.