ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC VÀ KỸ THUẬT THÔNG TIN

BÙI THỊ NGỌC MAI

KHÓA LUẬN TỐT NGHIỆP

DỰ ĐOÁN CHUỖI HÀNH ĐỘNG LIÊN QUAN TỚI NHAU CỦA CON NGƯỜI

Predicting interdependent and periodic action sequences of human

KỸ SƯ/ CỬ NHÂN NGÀNH CÔNG NGHỆ THÔNG TIN

TP. HỒ CHÍ MINH, 2021

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC VÀ KỸ THUẬT THÔNG TIN

BÙI THỊ NGỌC MAI – 17520731

KHÓA LUẬN TỐT NGHIỆP
DỰ ĐOÁN CHUỖI HÀNH ĐỘNG LIÊN QUAN TỚI NHAU CỦA CON NGƯỜI

Predicting interdependent and periodic action sequences of human

KỸ SƯ/ CỬ NHÂN NGÀNH CÔNG NGHỆ THÔNG TIN

GIẢNG VIÊN HƯỚNG DẪN
THS. NGUYỄN THỊ ANH THƯ

TP. HỒ CHÍ MINH, 2021

DANH SÁCH HỘI ĐỒNG BẢO VỆ KHÓA LUẬN

Hội đồng chấm khóa luận tốt nghiệp, thành lập theo Quyết định số …… ngày …….. của Hiệu trưởng Trường Đại học Công nghệ Thông tin.

1. ……. – Chủ tịch.
2. ……. – Thư ký.
3. ……. – Ủy viên.
4. ……. – Ủy viên.

LỜI CẢM ƠN

Đầu tiên em xin chân thành cảm ơn Ban giám hiệu, quý Thầy cô của trường Đại học Công nghệ Thông tin, đặc biệt là các thầy cô trong khoa Khoa Khoa học và Kỹ thuật thông tin đã tận tình giảng dạy, trang bị cho em những kiến thức cần thiết trong suốt những năm học tập tại trường.

Em xin chân thành gửi lời cảm ơn đến cô ThS. Nguyễn Thị Anh Thư vì đã tận tình hướng dẫn, giúp đỡ em trong suốt thời gian thực hiện khoá luận để em có thể hoàn thành khoá luận này một cách thuận lợi nhất.

Em cũng xin gửi lời cảm ơn đến gia đình, bạn bè lớp CNTT2017 đã giúp đỡ, đồng hành trong suốt thời gian học tập và sinh hoạt trong ở mái trường Đại Học.

Mặc dù đã cố gắng hoàn thành khoá luận với tất cả sự nỗ lực, nhưng khoá luận chắc chắn không tránh khỏi những thiếu sót, em kính mong quý thầy cô tận tình chỉ bảo.

Em xin chân thành cảm ơn!

TP. Hồ Chí Minh, tháng 08 năm 2021

Sinh viên thực hiện

Bùi Thị Ngọc Mai

MỤC LỤC

TÓM TẮT KHÓA LUẬN 1

Chương 1. MỞ ĐẦU 2

1.1. Giới thiệu 2

1.2. Phát biểu bài toán 3

1.3. Ứng dụng 2

1.4. Thách thức 2

1.5. Mục tiêu 3

1.6. Đối tượng và phạm vi nghiên cứu 3

1.7. Đóng góp của khoá luận 4

1.8. Cấu trúc của khoá luận 4

Chương 2. CÁC CÔNG TRÌNH NGHIÊN CỨU LIÊN QUAN 6

2.1. Dự đoán hành động tiếp theo 6

2.2. Kiến trúc lặp lại hành động 6

2.3. Quy trình điểm tạm thời 7

Chương 3. CƠ SỞ LÝ THUYẾT 8

3.1. Thuật toán Expectation – Maximization 8

3.2. Mô hình Gaussian mixture 8

3.3. Quy trình Hawkes 9

3.3.1. Quá trình tự kích thích 10

3.3.2. Hàm cường độ 10

3.4. Phân phối Weibull 11

3.5. Mô hình TIPAS 13
3.6. Mô hình Support Vector Classifier 13
3.7. Mô hình Logistic Regression 13
3.8. Mô hình phân lớp Naive Bayes 14
3.8.1. Định lý Bayes 14
3.8.2. Mô hình phân lớp Naive Bayes 14
3.9. Mô hình phân lớp K – Nearest Neighbor 16
Chương 4. MÔ HÌNH DỰ ĐOÁN 17
4.1. Giới thiệu 17
4.1.1. Ý tưởng 17
4.1.2. Thành phần dữ liệu và các định nghĩa 18
4.2. Mô hình dự đoán chuỗi hành động liên quan tới nhau của con người 19
4.2.1. Mô hình tổng quát 19
4.2.2. Phương pháp xây dựng mô hình 21
4.3. Kết luận 30
Chương 5. KẾT QUẢ THỰC NGHIỆM VÀ ĐÁNH GIÁ 31
5.1. Dữ liệu thực nghiệm 31
5.1.1. Độ phức tạp của dữ liệu 32
5.1.2. Cấu trúc phân bố dữ liệu 36
5.2. Độ đo đánh giá 37
5.2.1. Accuracy Score 37
5.2.2. Độ đo Macro Average Recall 38
5.3. Phương pháp thực nghiệm 39

5.3.1. Các phương pháp so sánh 39
5.3.2. Xử lý dữ liệu 40
5.3.3. Huấn luyện mô hình 43
5.4. Kết quả thực nghiệm 43
Chương 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 49
6.1. Kết quả đạt được 49
6.2. Hướng phát triển 50
TÀI LIỆU THAM KHẢO 51

DANH MỤC HÌNH VẼ

Hình 1-1 Mô hình hoá chuỗi các hành động liên quan tới nhau 3
Hình 3-1 Ví dụ hàm mật độ xác suất của phân phối Weibull 13
Hình 3-2 Ví dụ hàm mật độ xác suất của phân phối Weibull 13
Hình 4-1 Framework bài toán 21
Hình 4-2 Mô hình K – Nearest Neighbor tìm các điểm lân cận 28
Hình 4-3 Cách thức hoạt động của mô hình K – Nearest Neighbor 29
Hình 4-4 Ví dụ chọn giá trị K lân cận 30
Hình 5-1 Ma trận hỗn loạn của bộ dữ liệu 33
Hình 5-2 Thống kê tần số xuất hiện của các hành động 37
Hình 5-3 Thống kê tần số xuất hiện của các thời điểm xảy ra hành động 37
Hình 5-4 Ví dụ hành động người dùng 43
Hình 5-5 Kết quả tỷ lệ dự đoán chính xác khi dự đoán hành độngbằng độ đo Accuracy 44
Hình 5-6 Kết quả tỷ lệ dự đoán chính xác khi dự đoán hành động bằng độ đo Macro Average Recall 45
Hình 5-7: Kết quả tỷ lệ dự đoán chính xác khi dự đoán thời điểm xảy ra hành động bằng độ đo Accuracy 46
Hình 5-8 Kết quả tỷ lệ dự đoán chính xác khi dự đoán thời điểm xảy ra hành động bằng độ đo Macro Average Recall 47
Hình 5-9 Thời gian huấn luyện mô hình(giây) 48

DANH MỤC BẢNG

Bảng 4-1 Thành phần dữ liệu của bài toán dự đoán hành động và thời điểm xảy ra hành động 19
Bảng 5-1 Bảng thống kê số hành động của người dùng 36
Bảng 5-2 Ví dụ về độ đo Accuracy Score 38
Bảng 5-3 Các phương pháp so sánh 40

Bảng 5-4 Kết quả tỷ lệ dự đoán chính xác khi dự đoán hành động bằng độ đo Accuracy 44
Bảng 5-5 Kết quả tỷ lệ dự đoán chính xác khi dự đoán hành động bằng độ đo Macro Average Recall 45
Bảng 5-6 *Kết quả tỷ lệ dự đoán chính xác khi dự đoán thời điểm xảy ra hành động bằng độ đo Accuracy* 46
Bảng 5-7 Kết quả tỷ lệ dự đoán chính xác khi dự đoán thời điểm xảy ra hành động bằng độ đo Macro Average Recall 47
Bảng 5-8 Thời gian huấn luyện mô hình chuỗi hành động (giây) 49

DANH MỤC TỪ VIẾT TẮT

Từ viết tắt	**Nội dung**
LSTM	Long Short Term Memory
GMM	Gaussian Mixture Model
GD	Gaussian distribution
MLE	Maximum likelihood estimation
EM	Expectation Maximization
KNN	K – Nearest Neighbor
SVC	Support Vector Classifier
Gaussian NB	Gaussian Naive Bayes
a	Action
w	Window of times
t	Time
prd_a	Predict action
prd_w	Predict window of times
h	History
TP	True Positives
TN	True Negatives

FP	False Positives
FN	False Negatives

TÓM TẮT KHÓA LUẬN

Thực hiện các hoạt động như ăn, ngủ hay tập thể dục theo một lộ trình đã được lên kế hoạch có ý nghĩa rất quan trọng đối với sức khoẻ mỗi người. Tuy nhiên, với cuộc sống hiện nay, con người thường quá bận rộn với công việc của mình mà quên đi việc tập thể dục điều độ hoặc uống đủ 2 lít nước mỗi ngày. Thay vào đó, mọi người thường thực hiện các hoạt động cá nhân theo cảm hứng của mình và điều này có thể gây ảnh hưởng xấu đến sức khoẻ. Do đó, việc dự đoán và khuyến nghị hành động nên thực hiện vào từng thời điểm cụ thể là gì sẽ giúp con người cải thiện sức khoẻ mà không cần sự trợ giúp của những thực phẩm chức năng hay tư vấn của bác sĩ.

Đề tài được tiếp cận dựa trên những tính chất quan trọng của hành động đã được cá nhân hoá như xu hướng diễn ra hành động và thời gian các hành động xảy ra tương tự với quy trình điểm tạm thời. Sau đó, sử dụng mô hình TIPAS để dự đoán hành động sẽ xảy ra trong tương lai. Tiếp theo, kết hợp bộ dữ liệu cùng với kết quả hành động vừa dự đoán được cho vào mô hình *K – Nearest Neighbor* để dự đoán thời điểm xảy ra hành động đó của người dùng.

Đề tài được thực nghiệm trên bộ dữ liệu so sánh hơn 234.665 hành động của 295 người dùng trong hơn 17 tháng. Thông qua cách tiếp cận đề tài cùng bộ dữ liệu thực nghiệm, mô hình dự đoán của đề tài đã thành công cải thiện khả năng dự đoán các hành động và thời điểm hành động đó sẽ xảy ra trong tương lai. Cụ thể hơn, thông qua độ đo Accuracy Khoá luận đã nhận được kết quả cho công việc dự đoán hành động là 56,65% và công việc dự đoán thời điểm xảy ra hành động 62,82%. Bên cạnh đó, Khoá luận cũng đã giảm thời gian huấn luyện mô hình và dự đoán đáng kể khi dự đoán thời điểm xảy ra hành động. Điều này đã chứng minh rằng, mô hình dự đoán của đề tài có thể mô hình hoá hành vi người dùng và từ đó đề ra lộ trình hoạt động phù hợp để cải thiện sức khoẻ.

Ngoài ra, đề tài của chúng tôi đã được chấp nhận tại Hội nghị Khoa học Trường Đại học Khoa học tự nhiên – Đại học quốc gia thành phố Hồ Chí Minh lần thứ 12, vào tháng 12 năm 2020.

Từ khoá: Predict action – dự đoán hành động, data mining – khai thác dữ liệu, data science – khoa học dữ liệu, social network – mạng xã hội, K – Nearest Neighbor.

Chương 1. MỞ ĐẦU

1.1. Giới thiệu

Bảo vệ sức khoẻ đang là một vấn đề được hầu hết mọi người trên thế giới quan tâm. Minh chứng cho việc này chính là doanh số của các ứng dụng ghi lại hoạt động người dùng và thiết bị theo dõi sức khoẻ như đồng hồ thông minh liên tục tăng cao trong những năm trở lại đây. Vì thế, có thể thấy rằng, con người đang hình thành thói quen theo dõi sức khoẻ của mình và dựa trên những gợi ý của các thiết bị thông minh để thực hiện các hành động gắn liền với cuộc sống hàng ngày như tập thể dục, chế độ ăn uống, ngủ,…

Do đó, nếu có thể dự đoán chính xác chuỗi các hành động sẽ xảy ra trong tương lai có vai trò quan trọng với việc làm cho các ứng dụng và thiết bị ghi nhật kí hoạt động trở nên hữu ích hơn. Bên cạnh đó, bằng cách cung cấp cho người dùng trải nghiệm cá nhân hóa phù hợp với mục tiêu của họ sẽ có khả năng cải thiện sức khỏe một cách đáng kể. Thêm vào đó, khi người dùng thực hiện đúng với những hành động được gợi ý thì còn có thể ngăn chặn những tác nhân tiêu cực ảnh hưởng tới sức khỏe từ bên ngoài và thúc đẩy duy trì các hoạt động lành mạnh.

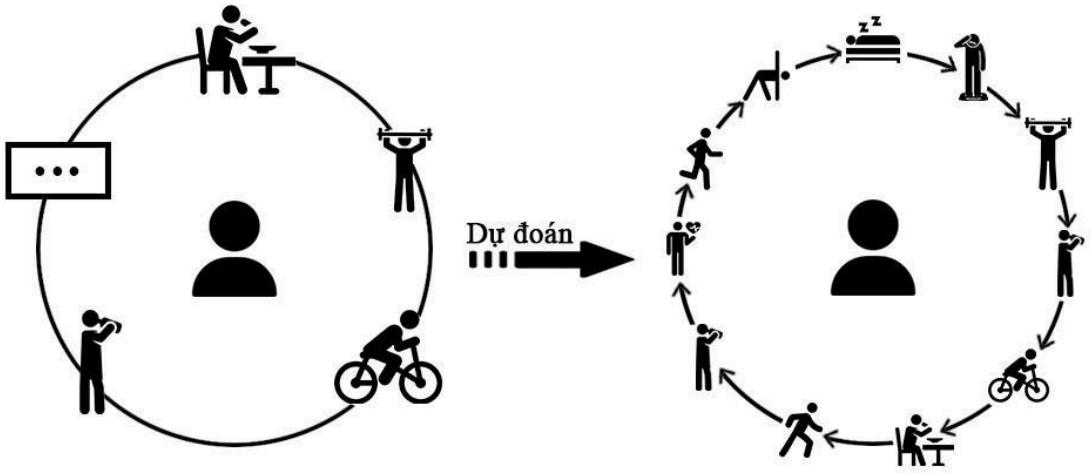
Ngoài ra, thông qua nhật ký hoạt động của người dùng để dự đoán những hành động thích hợp xảy ra sẽ tạo điều kiện cho các ứng dụng và thiết bị theo dõi sức khoẻ cá nhân hoá giao diện người dùng. Từ đó, người dùng sẽ được tự động cung cấp các thông tin và lời nhắc thực hiện hành động mà họ cần nhất mà không cần thực hiện bất kì thao tác nào khác. Vậy nên, thông qua những lời nhắc nhở đều đặn này, người dùng sẽ dễ nhận ra mục tiêu về sức khoẻ cá nhân hơn. Ví dụ, nhắc nhở người dùng đo cân nặng và tập thể dục vào mỗi buổi sáng có thể khiến họ hình thành ý thức giảm cân hay rèn luyện cơ thể. Nói một cách tổng quan hơn, có thể thấy việc dự đoán các hành động và thời điểm hành động đó sẽ xảy ra trong tương lai có vai trò như một người trợ lý dành riêng cho người dùng và người trợ lý này sẽ có nhiệm vụ khuyến nghị và hỗ trợ người dùng thực hiện đúng những hoạt động có lợi cho sức khoẻ.

1.2. Phát biểu bài toán

Cho trước bộ dữ liệu người dùng gồm các hoạt động hàng ngày như ăn, ngủ, đạp xe, đi bộ,… Làm thế nào để dự đoán các hành động và thời điểm xảy ra hành động đó trong tương lai và phù hợp với ngữ cảnh nhất.

Cụ thể hơn, bài toán Dự đoán chuỗi hành động liên quan tới nhau của con người được định nghĩa với input và output như sau:

- ***Input***: bộ dữ liệu gồm các tham số đầu vào là người dùng, các hành động của người dùng đó, thời gian xảy ra hành động, thời điểm xảy ra hành động. Trong đó:
    - Hành động của người dùng được phân thành 10 nhãn: uống, ngủ (thức dậy), nhịp tim, chạy bộ, cân nặng, thức ăn, đi bộ, đạp xe, tập thể dục, giãn cơ.
    - Thời điểm xảy ra hành động được chia thành 4 khung giờ: 0 – 6 giờ, 6 – 12 giờ, 12 – 18 giờ, 18 – 24 giờ.
- ***Output***: hành động và thời điểm hành động đó sẽ xảy ra trong tương lai.



Input: Các hoạt động hàng ngày của người dùng

Output: Các hoạt động và thời điểm xảy ra hành động đó của người dùng trong tương lai

Hình 1-1 Mô hình hoá chuỗi các hành động liên quan tới nhau

1.3. Ứng dụng

Nhờ vào tỷ lệ dự đoán chính xác hành động và thời điểm xảy ra hành động đó cao, nên chúng ta có thể sử dụng kết quả dự đoán tích hợp vào các ứng dụng và thiết bị theo dõi sức khoẻ để gợi ý và nhắc nhở người dùng thực hiện đúng các hoạt động phù hợp. Từ đây, doanh thu của các thiết bị theo dõi sức khoẻ có thể được tăng trưởng. Cụ thể hơn, đề tài có thể giải quyết một số bài toán sau:

- Hệ thống khuyến nghị trong các ứng dụng và thiết bị chăm sóc sức khoẻ người dùng.
- Hỗ trợ chăm sóc sức khoẻ cho bệnh nhân và đặc biệt đối với những người thiếu nhận thức hoặc những người không có khả năng tự chăm sóc bản thân.
- Dự đoán các hành động và thời điểm xảy ra hành động sao cho phù hợp với chuẩn an toàn sức khoẻ.
- Giúp người dùng hình thành thói quen rèn luyện cơ thể thông qua việc sinh hoạt điều độ và thường xuyên tập thể dục.

1.4. Thách thức

Hành vi của con người cực kỳ phức tạp, chúng thường xuyên thay đổi theo thời gian và bị các yếu tố khách quan bên ngoài ảnh hưởng rất lớn. Bên cạnh đó, các hành động của con người trong cuộc sống hàng ngày còn có sự thay đổi theo từng chu kì dài hạn và ngắn hạn. Ví dụ như, uống nước trước khi đi ngủ, ăn trưa vào lúc 12 giờ và ăn tối lúc 19 giờ. Tuy nhiên, giả thuyết được đặt ra rằng, nếu người dùng ăn trưa sớm hơn dự định thì có ăn tối sớm hơn không? Vì thế, vấn đề mô hình hoá các hành động và dự đoán chuỗi những hành động sẽ xảy ra trong tương lai là một công việc đầy thách thức.

Bên cạnh đó các kỹ thuật mô hình hoá và dự đoán hiện nay không thể đồng thời dự đoán chính xác tất cả các tính chất của hành động (thay đổi theo thời gian, liên quan lẫn nhau và có định kỳ). Thế nhưng, nếu không xét hết tất cả các tính chất này thì tỷ lệ dự đoán sẽ rất thấp. Bên cạnh đó, bài toán được đặt ra đối với đề tài là bài toán phân lớp, đặc biệt đối với công việc dự đoán hành động sẽ xảy ra trong tương lai thì chúng tôi có 10 lớp cần phân

loại. Vì thế, việc dự đoán chính xác các hành động và thời điểm xảy ra hành động trong tương lai là một thách thức rất lớn.

Ngoài ra, hầu hết các thông tin hoạt động của người dùng trên các ứng dụng theo dõi sức khoẻ đều có độ bảo mật rất cao do vấn đề riêng tư. Vậy nên, những thông tin như giới tính, độ tuổi, tôn giáo, dân tộc,… là những yếu tố có ảnh hưởng lớn tới hành vi hoạt động của con người nhưng không được đưa vào bộ dữ liệu. Cũng chính vì lý do này mà bộ dữ liệu chỉ có kích thước giới hạn và là bộ dữ liệu không cân bằng. Điều này cũng ảnh hưởng rất lớn tới quá trình huấn luyện mô hình dự đoán.

1.5. Mục tiêu

- Nghiên cứu cải tiến phương pháp dự đoán chuỗi các hành động và thời điểm xảy ra các hành động đó trong tương lai, giúp nâng cao độ chính xác dự đoán.
- Áp dụng các thuật toán máy học vào phương pháp mô hình hóa các chuỗi hành động liên quan tới nhau trong thế giới thực.

1.6. Đối tượng và phạm vi nghiên cứu

- Phạm vi nghiên cứu: Đề tài được thực nghiệm trên bộ dữ liệu so sánh 234.665 hành động được thực hiện bởi 295 người dùng trong hơn 17 tháng (Do vấn đề quyền riêng tư và bảo mật nên dữ liệu được trích ra từ bộ dữ liệu chính gồm 12 triệu hành động của 20.000 người dùng và được mã hoá các thông tin người dùng & hành động của người dùng).
- Đối tượng nghiên cứu: Sử dụng tập dữ liệu về các các hoạt động đã ghi từ ứng dụng hoạt động trên thiết bị di động là Argus và Under Armour (Dữ liệu được công khai bởi đại học Stanford). Người dùng trong bộ dữ liệu có 10 loại hành động khác nhau: uống, ngủ, nhịp tim, chạy, cân nặng, thức ăn, đi bộ, đạp xe, tập thể dục và giãn cơ. Các hành động này được thực hiện theo 4 khung giờ trong ngày: 0 giờ đến 6 giờ, 6 giờ đến 12 giờ, 12 giờ đến 18 giờ, 18 giờ đến 24 giờ.

  Link download data: http://snap.stanford.edu/tipas/

1.7. Đóng góp của khoá luận

Sau khi phân tích tính chất của bộ dữ liệu cũng như đặc điểm các hành động thường ngày của con người. Chúng tôi đã thực hiện được những công việc sau:

- Đơn giản hoá bộ dữ liệu của người dùng.
- Sử dụng mô hình máy học để dự đoán thành công hành động và thời điểm xảy ra hành động người dùng.
- Cải thiện thời gian dự đoán đáng kể so với các phương pháp baseline.
- Cải thiện độ chính xác của mô hình dự đoán baseline.
- Đề tài được chấp nhận tại Hội nghị Khoa học trường Đại học Khoa học tự nhiên – Đại học quốc gia thành phố Hồ Chí Minh lần thứ 12, vào tháng 12/2020.

1.8. Cấu trúc của khoá luận

Bố cục của khóa luận bao gồm 06 phần với nội dung chính như sau:

Chương 1: Mở đầu

Trình bày khái quát và phát triển bài toán dự đoán chuỗi hành động liên quan tới nhau của con người. Đồng thời cũng trình bày thực trạng, những khó khăn, thách thức và ứng dụng của bài toán. Từ đó xác định mục tiêu, đối tượng và phạm vi nghiên cứu cho khóa luận.

Chương 2: Các công trình nghiên cứu liên quan

Trình bày, phân tích và đánh giá các hướng tiếp cận đã được công bố liên quan đến bài toán dự đoán chuỗi hành động liên quan tới nhau của con người. Từ những công trình nghiên cứu liên quan, xác định và xây dựng các giải pháp cho những vấn đề còn tồn tại và thách thức trong bài toán.

Chương 3: Cơ sở lý thuyết

Trình bày chi tiết những cơ sở lý thuyết cần thiết để xây dựng mô hình dự đoán chuỗi hành động liên quan tới nhau của con người, bao gồm:

- Thuật toán Expectation Maximization
- Mô hình Gaussian
- Quy trình Hawkes
- Phân phối Weibull

- Mô hình TIPAS
- Mô hình Support Vector Classifier
- Mô hình Logistic Regression
- Mô hình phân lớp Gaussian Naive Bayes
- Mô hình phân lớp K – Nearest Neighbor

Chương 4: Mô hình dự đoán chuỗi hành động liên quan tới nhau của con người

Trình bày chi tiết mô hình dự đoán chuỗi hành động liên quan tới nhau của con người, trong đó bao gồm 3 quá trình:

- Xử lý dữ liệu
- Huấn luyện mô hình
- Dự đoán chuỗi hành động liên quan tới nhau của con người

Chương 5: Kết quả thực nghiệm và đánh giá

Trình bày các kết quả thực nghiệm của phương pháp đề xuất. Tiến hành so sánh, đánh giá và phân tích với các kết quả thực nghiệm của các phương pháp cơ sở đã được công bố thông qua độ đo đánh giá Prediction Accuracy Score và Macro Average Recall.

Chương 6: Kết luận và hướng phát triển

Kết luận quá trình nghiên cứu và thực nghiệm. Từ đó, đưa ra những định hướng nghiên cứu mở rộng cho đề tài trong tương lai.

Chương 2. CÁC CÔNG TRÌNH NGHIÊN CỨU LIÊN QUAN

Hiện nay đã có rất nhiều công trình nghiên cứu về dự đoán chuỗi hành động xảy ra trong quá khứ. Một số mô hình thường được sử dụng trong những công trình này là Markov, LTSM và phân phối Weibull. Bên cạnh đó, bởi tính chất phức tạp của hành động mà những công trình liên quan thường kết hợp nhiều thuật toán và phân phối xác suất với nhau [1]. Cụ thể hơn, các đề tài nghiên cứu liên quan được chia theo 3 nhóm chính như sau:

2.1. Dự đoán hành động tiếp theo

Công trình dự đoán hành động tiếp theo đã được nghiên cứu và thực hiện trên nhiều bài toán như dự đoán hành vi người dùng để xây dựng giao diện phù hợp [2], hay dự đoán những hành vi của người dùng khi tuỳ chọn mục sản phẩm [3], mua hàng trực tuyến [4],… Những công trình này thường sử dụng mô hình Markov nhằm dự đoán một chuỗi hành động có thời gian xảy ra rời rạc. Tuy nhiên, chính bởi tính rời rạc này mà mô hình Markov khi dự đoán những hành động bị ảnh hưởng bởi các chu kì dài hạn sẽ có tỷ lệ chính xác không cao [5]. Bên cạnh đó, các công trình nghiên cứu về bài toán dự đoán hành động tiếp theo còn thường sử dụng mô hình LSTM [6]. Mô hình này cũng có chức năng giả định các khoảng thời gian xảy ra rời rạc và thông qua những dữ liệu đã xảy ra trong quá khứ hay còn được gọi là lịch sử hoạt động của người dùng mà dự đoán các hành động tiếp theo. Mặc dù đã được cải tiến từ mạng nơ – ron hồi quy (Recurrent Neural Network – RNN), nhưng nhược điểm của LSTM vẫn là bị giới hạn khả năng ghi nhớ trong khoảng thời gian dài hạn.

Nhìn chung, mặc dù đã có rất nhiều công trình nghiên cứu về đề tài dự đoán hành động tiếp theo của con người. Thế nhưng, về lĩnh vực dự đoán hành động hàng ngày như ăn, uống, ngủ, nghỉ,… của con người thì chưa nhiều. Bên cạnh đó, các công trình này thường chỉ dự đoán hành động sẽ xảy ra tiếp theo là hành động nào mà không dự đoán thời điểm sẽ xảy ra. Đặc biệt, bộ dữ liệu thực nghiệm của đề tài ghi lại hành động của con người trong thế giới, vì thế các hành động này sẽ có sự ảnh hưởng bởi các chu kỳ dài hạn. Vì thế, đề tài của chúng tôi đã giải quyết đồng thời 2 bài toán này với độ phức tạp của hành động

lớn hơn và vấn đề giới hạn khả năng ghi nhớ dữ liệu của mô hình LSTM cũng đã được mô hình đề tài sử dụng giải quyết.

2.2. Kiến trúc lặp lại hành động

Con người thường có thói quen lặp lại hành động đã thực hiện trước đó. Ví dụ như, một người đi trên một con đường thì có khả năng cao người đó sẽ đi lại con đường ấy trong tương lai gần. Tổng quan hơn, hành vi của con người thường thay đổi theo thời gian nhưng lại có sự phụ thuộc lẫn nhau và diễn ra theo một chu kì nhất định.

Những công trình đã nghiên cứu về công trình kiến trúc lặp lại hành động thường tập trung vào các hành vi tiêu thụ sản phẩm của người dùng như xem video cầu nối [7], nghe nhạc [8], tìm kiếm thông tin trên trang web [9],… Những công trình này đã mô hình hoá hành động của nhằm mục đích nghiên cứu tính chất lặp lại hành động cũ và thực hiện các hành động mới của con người.

Ngoài ra, những công trình về kiến trúc lặp lại hành động còn thực hiện những khảo sát quan trọng của các đối tượng người dùng khác nhau để có một cái nhìn tổng quan và khái quát nhất. Cũng nhờ những nghiên cứu thông qua khảo sát và chúng ta có thể kết luận rằng, hành động của con người trong thế giới thực có tính chất lặp lại hành động cũ như một thói quen. Thêm vào đấy, thời gian hành động này được lặp lại sẽ khác nhau và bị ảnh hưởng vào tính chất của từng hành động cụ thể.

Tuy nhiên, hầu hết các hoạt động này không bị ảnh hưởng bởi cái chu kì ngắn hạn hay dài hạn khác nhau và thường ít thay đổi theo thời gian. Điều này trái lại với mục tiêu xây dự mô hình dự đoán của đề tài. Tuy nhiên, những công trình này là tiền đề để phát triển quá trình mô hình hoá hành động dựa trên tính chất lặp lại hành động theo thói quen của người dùng.

2.3. Quy trình điểm tạm thời

Đối với công trình nghiên cứu quy trình điểm tạm thời, các đề tài thường xoay quanh vấn đề dự đoán thời gian diễn ra các hành động thay đổi tiếp theo [10]. Từ đó, thông qua mô hình Poisson và quy trình Hawkes [11] mô hình hoá các ảnh hưởng của người dùng trong mạng xã hội [12] [13] [14]. Sau đó, phân tích sự phát triển của thông tin và cấu trúc mạng [15], phân cụm tài liệu [16], thời gian người dùng quay lại thực hiện hành động cũ [17],…

Với mô hình Poisson, đây là một mô hình hồi quy và sử dụng phương thức phân phối xác suất rời rạc để huấn luyện mô hình. Khác với những mô hình xác suất thông thường, Poisson không dựa vào xác suất các sự kiện xảy ra hay số lần xuất hiện của sự kiện. Thay vào đó, Poisson sẽ tính số lần trung bình hành động đó xảy ra trong một khoảng thời gian nhất định. Còn đối với quy trình Hawkes, còn được xem như một quy trình tự kích thích và quy trình điểm. Vì thế, các sự kiện sắp xảy ra của nó sẽ có sự phụ thuộc vào các sự kiện trước đó.

Bài toán Dự đoán chuỗi hành động liên quan tới nhau của con người đã mở rộng phạm vi nghiên cứu của công trình này và sử dụng phân phối mũ và Weibull kernels để mô hình hoá xu hướng thay đổi hành động theo thời gian của con người. Qua đó, chứng minh rằng tính chất này rất quan trọng khi dự đoán hành động và thời điểm xảy ra hành động của người dùng.

Do đó, có thể thấy rằng những công trình nghiên cứu về quy trình điểm tạm thời có ý nghĩa rất quan trọng đối với đề tài và đây cũng là tiền đề để đề tài có thể quyết định lựa chọn mô hình phù hợp với yêu cầu bài toán và bộ dữ liệu phức tạp.

Chương 3. CƠ SỞ LÝ THUYẾT

3.1. Thuật toán Expectation – Maximization

Expectation - Maximization là thuật toán thống kê để tìm tham số của phân phối xác suất. Thuật toán này còn là một phương pháp để tìm giá trị MLE (ước tính khả năng xảy ra tối đa) và thường được sử dụng khi dữ liệu không đầy đủ (bị missing data) [18]. Ưu điểm của thuật toán này là đồng thời tối ưu hoá các tham số của mô hình và có khả năng dự đoán các điểm dữ liệu bị thiếu (missing data) trong tất cả các vòng lặp.

Thuật toán EM bao gồm 3 tiến trình như sau: [19]

- E – step: ban đầu thuật toán sẽ khởi tạo các tham số một cách ngẫu nhiên cho các phân phối xác suất. Sau đó sử dụng các tham số này để tính xác suất điểm dữ liệu thuộc một phân phối xác suất cụ thể. Qua đó có thể gán được từng điểm dữ liệu vào một phân phối xác suất cụ thể.
- M – step: sử dụng các điểm dữ liệu đã được gán để cập nhật tham số của các phân phối xác suất.
- Lặp lại bước E – step và M – step cho tới khi các tham số của mô hình ổn định hoặc hội tụ.

3.2. Mô hình Gaussian mixture

Gaussian mixture là mô hình xác suất giả định tất cả các điểm dữ liệu được tạo ra từ một tập hữu hạn k các phân phối xác suất chuẩn (Gaussian distribution - GD) chưa biết tham số. Thuật toán này xem mỗi cụm là một phân phối xác suất chuẩn, k GD tương ứng với k cụm [20]. Đối với một tập dữ liệu cụ thể, mô hình Gaussian mixture sẽ xác định xác suất điểm dữ liệu thuộc một phân phối xác suất cụ thể. Các tham số GMM được đánh giá từ việc huấn luyện dữ liệu sử dụng thuật toán Expectation Maximization – EM.

Mô hình hỗn hợp Gaussian là tổng k của các mật độ phân phối Gaussian, trong đó k ∈ {1,…, K} và K là số lượng phân cụm của bộ dữ liệu (trong bộ dữ liệu thực nghiệm của đề tài có 10 phân cụm hành động người dùng). Mỗi thành phần k Gaussian trong hỗn hợp Gaussian chứa các tham số sau [21]:

- μ: giá trị trung bình của phân phối
- Σ: hiệp phương sai, định nghĩa kích thước của một phân phối
- π: giá trị xác suất của phân phối

Cụ thể hơn, mô hình hỗn hợp Gaussian có công thức sau:

$$p(x) = \sum_{k=1}^{k} + \pi_k N(x|\mu_k, \sum k) \quad (1)$$

3.3. Quy trình Hawkes

Quy trình Hawkes được xem như một quy trình tự kích thích và quy trình điểm. Vì thế, các sự kiện sắp xảy ra của nó sẽ có sự phụ thuộc vào các sự kiện trước đó.

Nếu chúng ta gọi quy trình điểm tạm thời tự kích thích là *N* thì hàm điều kiện cường độ là λ = λ(t) được định nghĩa như sau [22]:

$$(t) = (t) + \sum_{i:\ _i<t} (t - _i) \quad (2)$$

Trong đó:

- (t) là tỷ lệ xảy ra quy trình *N*
- $_i$ là điểm thời gian xảy ra sự kiện trước đó tới thời gian t
- là hàm chi phối mật độ phân cụm của *N* hay hàm kích thích của *N*

Cụ thể hơn, quy trình Hawkes sẽ liên quan tới 2 vấn đề là quá trình tự kích thích và hàm cường độ.

3.3.1. Quá trình tự kích thích

Quá trình tự kích thích bản chất là một quy trình điểm, trong đó khi một sự kiện xuất hiện sẽ làm tăng điều kiện của hàm cường độ. Do đó, quá trình tự kích thích còn có sự phụ thuộc vào quy trình đếm (counting process) và những sự kiện đã xảy ra trước đó đối với hàm cường độ của quy trình đếm. Thông qua quá trình tự kích thích, quy trình Hawkes được định nghĩa như sau [23]:

"Gọi $\{N_t\}_{t>0}$ là quy trình đếm với lịch sử hoạt động của các sự kiện liên quan là H_t, $t \geq 0$. Quy trình điểm được xác định bởi hàm cường độ của các sự kiện $\lambda(t)$ và quy trình này sẽ trở thành Hawkes nếu hàm cường độ điều kiện được tính theo công thức sau:

$$\lambda(H_t) = \lambda_0(t) + \sum_{i:t>T_i} \phi(t - T_i) \quad (3)$$

Trong đó, $T_i < t$ là tất cả các sự kiện thời gian diễn ra trước thời điểm t hiện tại và những sự kiện này sẽ tác động tới hàm cường độ tại thời điểm t. Bên cạnh đó, $\lambda_0(t)$ là yếu tố quyết định nền tảng của hàm cường độ và ϕ được gọi là bộ nhớ kernel."

3.3.2. Hàm cường độ

Đại lượng $\lambda_0(t) > 0$ là điều kiện cần của hàm cường độ, chúng diễn tả sự xuất hiện của các sự kiện đã thoả điều kiện triggers bởi external sources. Mặc dù những sự kiện này được xem như các sự kiện "nhập cư" và chúng xảy ra độc lập so với những sự kiện trước đó trong quy trình, nhưng chúng lại là yếu tố tác động tới giá trị của quá trình tự kích thích. Khi kernel $\phi(t - T_i)$ thay đổi sự kiện tại thời điểm T_i trong hàm cường độ tại thời gian t thì giá trị của hàm tự kích thích sẽ tăng lên [23]. Thông thường, hàm $\phi(\cdot)$ có chức năng giảm sự đơn điệu của sự kiện để các sự kiện vừa mới xảy ra có ảnh hưởng lớn hơn đến sự kiện đang xảy ra so với các sự kiện đã xảy ra cách đó lâu hơn. Do đó, chúng ta có thể thấy rằng hàm cường độ sẽ ảnh hưởng tới khả năng xảy ra hành động trong một chu kì ngắn hạn và điều này sẽ giúp kết quả dự đoán của đề tài chính xác hơn.

3.4. Phân phối Weibull

Phân phối Weibull là loại phân phối liên tục và được áp dụng trong nhiều lĩnh vực khác nhau như khoa học, y học, năng lượng, điện năng, thuỷ năng,… Đặc biệt, hàm phân phối Weibull rất phù hợp với các bộ dữ liệu có kích thước lớn và giá trị không đồng nhất. Vì thế, chúng ta có thể áp dụng phân phối Weibull vào tính toán sự thay đổi của các sự kiện độc lập và có tính chu kỳ trong khoảng thời gian dài hạn [24].

Hàm phân phối Weibull là hàm phân phối liên tục. Hàm mật độ xác suất và hàm phân phối xác suất của hàm Weibull như sau [25]:

$$f(x) = \frac{\alpha}{\beta}\left(\frac{x}{\beta}\right)^{\alpha-1} e^{-\left(\frac{x}{\beta}\right)^{\alpha}} \tag{4}$$

$$F(x) = 1 - e^{-\left(\frac{x}{\beta}\right)^{\alpha}} \tag{5}$$

Trong đó α và β là các số không âm và lần lượt là tham số hình dáng và tỉ lệ của hàm Weibull.

Giá trị trung bình (kỳ vọng) và phương sai được tính như sau:

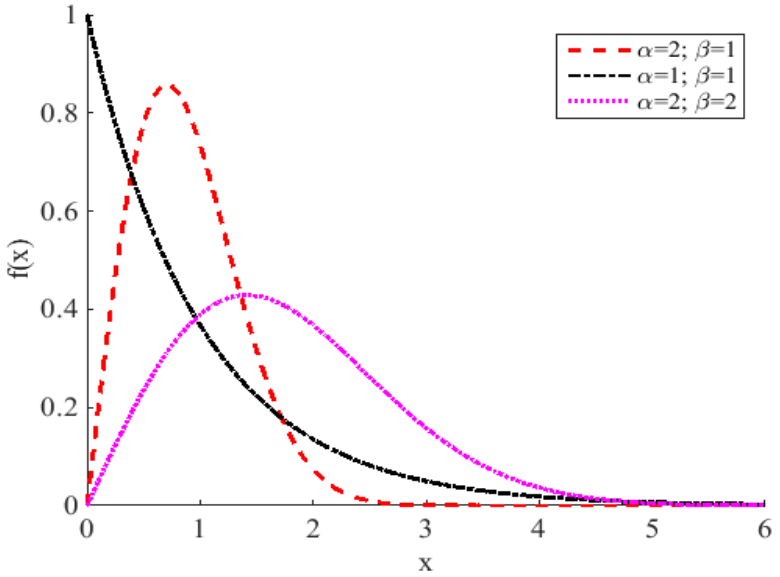
$$\mu = \beta\Gamma\left(1 + \frac{1}{a}\right) \tag{6}$$

$$\sigma^2 = \beta\Gamma\left(1 + \frac{2}{a}\right) - \left[\beta\Gamma\left(1 + \frac{1}{a}\right)\right]^2 \tag{7}$$

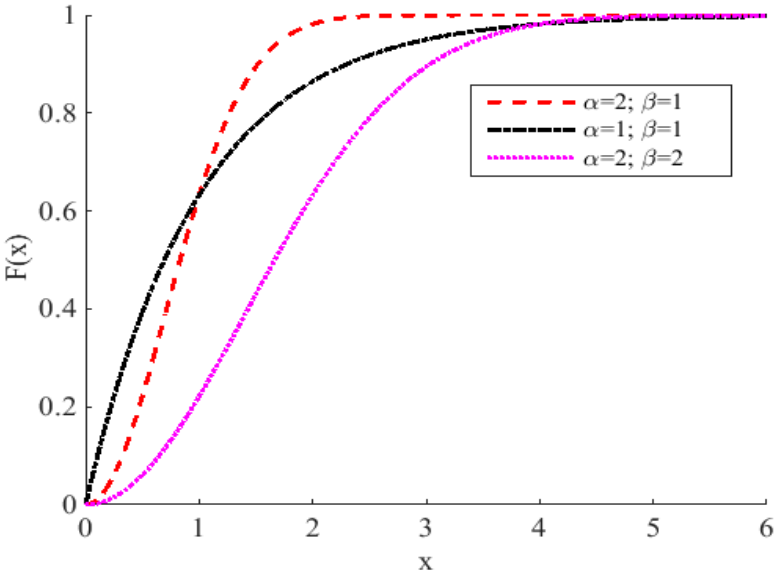
Trong đó $\Gamma(\cdot)$ là hàm nổi tiếng Gamma trong xác suất và thống kê. Đối với một số thực không âm r, hàm được tính như sau:

$$\Gamma(r) = \int_0^{\infty} t^{r-1} e^{-t} dt\,. \tag{8}$$

Sau đây là ví dụ về hàm mật độ xác suất và hàm phân phối xác suất của phân phối Weibull.



Hình 3-1 Ví dụ hàm mật độ xác suất của phân phối Weibull



Hình 3-2 Ví dụ hàm mật độ xác suất của phân phối Weibull

3.5. Mô hình TIPAS

Mô hình TIPAS được xây dựng bởi 3 tác giả thuộc trường đại học Stanford là: Takeshi Kurashima, Tim Althoff, Jure Leskovec vào tháng 4 năm 2018. Mô hình này được xây dựng dựa trên các tính chất đặc trưng của hành động người dùng trong tương lai:

- Thay đổi hành động theo thời gian
- Thực hiện hành động theo thói quen
- Các hành động có tính độc lập với nhau
- Bị ảnh hưởng bởi các chu kỳ ngắn hạn và dài hạn khác nhau

Cùng với những tính chất này, mô hình TIPAS còn được kết hợp mô hình hỗn hợp Gaussian, quy trình Hawkes và phân phối Weibull với thuật toán EM để tính toán các tham số mô hình. Từ đó, thực hiện quá trình huấn luyện và dự đoán chuỗi hành động sẽ xảy ra trong tương lai.

3.6. Mô hình Support Vector Classifier

Support Vector Classifier hay còn gọi là SVC, SVC được phát triển dựa trên mô hình Support Vector Machine (SVM) – mô hình máy học mạnh mẽ nhất trong lĩnh vực Machine Learning. Vì thế, SVC là mô hình máy học khá phổ biến trong Machine Learning. Mô hình này có thể sử dụng cho các bài toán phân lớp và đặc biệt phù hợp với mọi loại dữ liệu của các bài toán, dù là dữ liệu lớn, nhỏ, đơn giản hay phức tạp [26]. Hơn nữa, mô hình SVC còn có thể giải quyết bài toán phân lớp nhị phân và phân lớp đa biến.

3.7. Mô hình Logistic Regression

Thuật toán Logistic Regression là một thuật toán hồi quy nhưng nó có thể sử dụng cho các bài toán phân lớp. Cách thức hoạt động của thuật toán này là ước tính xác suất một trường hợp nào đó rơi vào lớp cụ thể nào. Đối với bài toán dự đoán đa biến, thuật toán này sẽ tính xác suất của tất cả các trường hợp và chọn lớp có xác suất cao nhất.

$$P(x; w);\ P(y = 0|x; w) \tag{8}$$

Tuy nhiên, nhược điểm của mô hình này là đòi hỏi các điểm dữ liệu huấn luyện phải là được tạo ra độc lập với nhau. Thế nhưng, dữ liệu thực tế thì thường có mối tương quan lẫn

nhau. Do đó, thuật toán Logistic Regression chỉ phù hợp với một số bộ dữ liệu nhất định. [27]

3.8. Mô hình phân lớp Naive Bayes

Mô hình phân lớp Naive Bayes được phát triển dựa trên định lý Bayes. Vì thế, bộ phân lớp Bayes có tính chất như một giải thuật xác suất thống kê. Thông qua phương thức tính toán, mô hình này sẽ tính ra xác suất một phần tử dữ liệu thuộc vào từng lớp. Từ đó, chọn ra lớp có xác suất cao nhất và phân lớp phần tử dữ liệu theo yêu cầu của người dùng.

3.8.1. Định lý Bayes

Ta có công thức Bayes như sau [28]:

$$P(B) = \frac{P(B|A)P(A)}{P(B)} \quad (9)$$

Công thức trên là một công thức xác suất có điều kiện, chỉ ra xác suất của A xảy ra nếu B cũng xảy ra, ta viết là P(A|B). Ngược lại, nếu biết xác suất của B xảy ra khi biết A thì ta có công thức sau:

$$P(A) = \frac{P(A|B)P(B)}{P(A)} \quad (10)$$

Trong đó:

- P(A|B) là xác suất của A khi biết B
- P(A) là xác suất xảy ra của A
- P(B|A) là xác suất của B khi biết A
- P(B) là xác suất xảy ra của B

3.8.2. Mô hình phân lớp Naive Bayes

Được bắt nguồn từ định lý Bayes, nên mô hình phân lớp Naive Bayes cũng có công thức toán học tương tự Bayes và chỉ có sự khác biệt về các thuật ngữ chuyên ngành và ý nghĩa phần tử.

Công thức [29]:

$$P(X) = \frac{P(X|y)P(y)}{P(X)} \quad (11)$$

Trong đó:

- y: Mục tiêu
- X: đặc trưng
- P(y|X) được gọi là posterior probability, đây là xác suất của y đối với X.
- P(X|y) được gọi là likelihood, đây là xác suất X đã biết tới y
- P(y) được gọi là prior probability của y
- P(X) được gọi là prior probability của X

Bên cạnh đó, X ở đây là một vector đặc trưng và có thể được biết lại như sau:

$$X = (x_1, x_2, x_3, ..., x_n) \quad (12)$$

Lúc này, đẳng thức Bayes sẽ trở thành:

$$P(x_1, x_2, ..., x_n) = \frac{P(y)P(y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)} \quad (13)$$

Kết quả của y sẽ phụ thuộc vào giá trị cực đại P(y|X) và giá trị này sẽ được tính toán như sau:

$$y = argmax_y P(y)\Pi_{i=1}^{n} P(x_i|y) \quad (14)$$

Đối với mô hình Naive Bayes, chúng ta cần lưu ý 2 đặc điểm sau:

- Các đặc trưng của mô hình có tính độc lập với nhau, nghĩa là khi ta thay đổi giá trị của một đặc trưng x_1 nào đó thì các đặc trưng $x_2, x_3, ..., x_n$ sẽ không bị ảnh hưởng tới.
- Đối với mục tiêu của output, các đặc trưng của mô hình có tính quan trọng ngang bằng nhau.

Bởi vì Bayes là một thuật toán mạnh mẽ và có hiệu quả lớn trong các bài toán, nên mô hình phân lớp Naive Bayes được ứng dụng rất rộng rãi. Một số ứng dụng của mô hình:

- Dự đoán với thời gian thực
- Phân loại văn bản
- Lọc thư rác
- Hệ thống recommendation

3.9. Mô hình phân lớp K – Nearest Neighbor

K - Nearest Neighbor hay còn được gọi là KNN, đây là một phương pháp phân lớp dựa trên thể hiện (Instance - based) [30]. Với phương pháp này, dữ liệu được sử dụng để huấn luyện và dự đoán sự kiện tiếp theo sẽ được lưu trữ và khi có yêu cầu xử lý phân lớp thì chúng mới được gọi lên xử lý. Mặc dù vấn đề này sẽ mất nhiều bộ nhớ và thời gian lưu trữ, tuy nhiên vì vậy nên quá trình dự đoán của mô hình này được diễn ra khá nhanh.

Phương thức hoạt động của phương pháp KNN là dựa vào phân lớp của các điểm lân cận để quyết định phân lớp của mình. Vì thế, ưu điểm của mô hình KNN là có độ phức tạp rất thấp (gần như bằng 0) và chúng ta cũng không cần đặt các giả sử về phân phối của class. Cũng chính vì thế mà huấn luyện và dự đoán mô hình diễn ra rất đơn giản.

Tuy nhiên, bên cạnh các ưu điểm thì mô hình KNN còn có những nhược điểm sau:

- Nếu K – Neighbor nhỏ thì sẽ rất nhạy cảm với bộ dữ liệu bị nhiễu.
- K càng lớn thì độ phức tạp của mô hình càng cao.
- Việc lưu trữ bộ dữ liệu trong bộ nhớ sẽ gây ảnh hưởng tới hiệu năng của KNN.

Chương 4. MÔ HÌNH DỰ ĐOÁN

4.1. Giới thiệu

4.1.1. Ý tưởng

Hầu hết các mô hình dự đoán phân lớp hay gom cụm thông thường chỉ có thể dự đoán các sự kiện xảy ra rời rạc, không có tính mật thiết với nhau. Tuy nhiên, các hành động của con người trong thế giới thực thì lại có tính chất cá nhân hoá và bị hành động xảy ra trong quá khứ sẽ có tác động rất lớn đối với hành động xảy ra tiếp theo. Vì thế, mô hình được sử dụng trong đề tài phải đồng thời giải quyết được các vấn đề này và mô hình TIPAS chính là minh chứng cho điều này.

Thêm vào đó, hành động của con người cũng có tính độc lập, không xảy ra theo một quy trình nhất định và có sự thay đổi trong một chu kỳ dài hạn. Điều này khiến bộ dữ liệu lưu trữ hành động có kích thước rất lớn và là tập dữ liệu nhiễu. Vì thế, mô hình được lựa chọn trong đề tài – KNN là một mô hình máy học phù hợp với tập dữ liệu này khi dự đoán thời điểm xảy ra hành động trong tương lai.

Bên cạnh đó, việc chọn giá trị K – neighbor cũng là một công việc hết sức quan trọng, quyết định tính hiệu quả của mô hình. Nếu K quá nhỏ thì sẽ nhạy cảm với tính nhiễu của dữ liệu, còn nếu K quá lớn thì vùng lân cận có khả năng cao chứa các điểm dữ liệu của nhãn khác. Vì thế, chúng tôi đã thông qua thực nghiệm để chọn giá trị K phù hợp.

4.1.2. Thành phần dữ liệu và các định nghĩa

Bài toán dự đoán hành động và thời điểm xảy ra hành động đó của người dùng được lấy dữ liệu từ những ứng dụng theo dõi hoạt động của người dùng. Do đó, bộ dữ liệu thực nghiệm của đề tài đảm bảo được tính chân thực và thể hiện rõ nét những tính chất về hành động của con người trong thế giới thực.

Cụ thể hơn, bộ dữ liệu gồm 5 thành phần chính như sau:

- ***a***: (*action*) id hành động người dùng
- ***t:*** (*time*) thời gian hành động ***a*** kéo dài, được tính bắt đầu từ thời điểm quan sát thực nghiệm cho tới thời điểm hành động được ghi lại và có đơn vị là phút.
- ***w***: (*window of time*) thời điểm diễn ra hành động ***a***
- ***prd_a***: id hành động tiếp theo của người dùng sau hành động ***a***
- ***prd_w*:** thời điểm diễn ra hành động ***prd_a***

a	t	w	prd_a	prd_w
1	7734.2	1	2	1
2	7734.63	1	0	1
0	7738.51	1	1	1
1	12085.47	1	1	1
2	449.33	1	1	1
1	4833.99	1	4	1
4	4867.23	1	4	1
0	5219.02	2	0	3
0	5606.45	3	0	0
0	6076.91	0	0	1
0	6316.37	1	1	1
0	557	1	2	1
2	557.2	1	0	1
0	1895.86	1	3	2
3	2209.42	2	0	3
…	…	…	…	…

Các dòng dữ liệu

Bảng 4-1 Thành phần dữ liệu của bài toán dự đoán hành động và thời điểm xảy ra hành động

Bên cạnh đó, mỗi dòng dữ liệu đại diện cho thông tin một hành động tại thời điểm ghi nhận (id hành động, thời gian hành động đó kéo dài và id thời điểm xảy ra hành động) và thông tin hành động được dự đoán sẽ xảy ra trong tương lai (id hành động và id thời điểm xảy ra hành động đó). Vì thế, ở những dòng dữ liệu có cùng id hành động và id thời điểm xảy ra hành động, thời gian hành động kéo dài sẽ được cộng dồn. Nói cách khác, thời gian kéo dài một hành động cụ thể nào đó sẽ được tăng lên theo thời gian.

4.2. Mô hình dự đoán chuỗi hành động liên quan tới nhau của con người

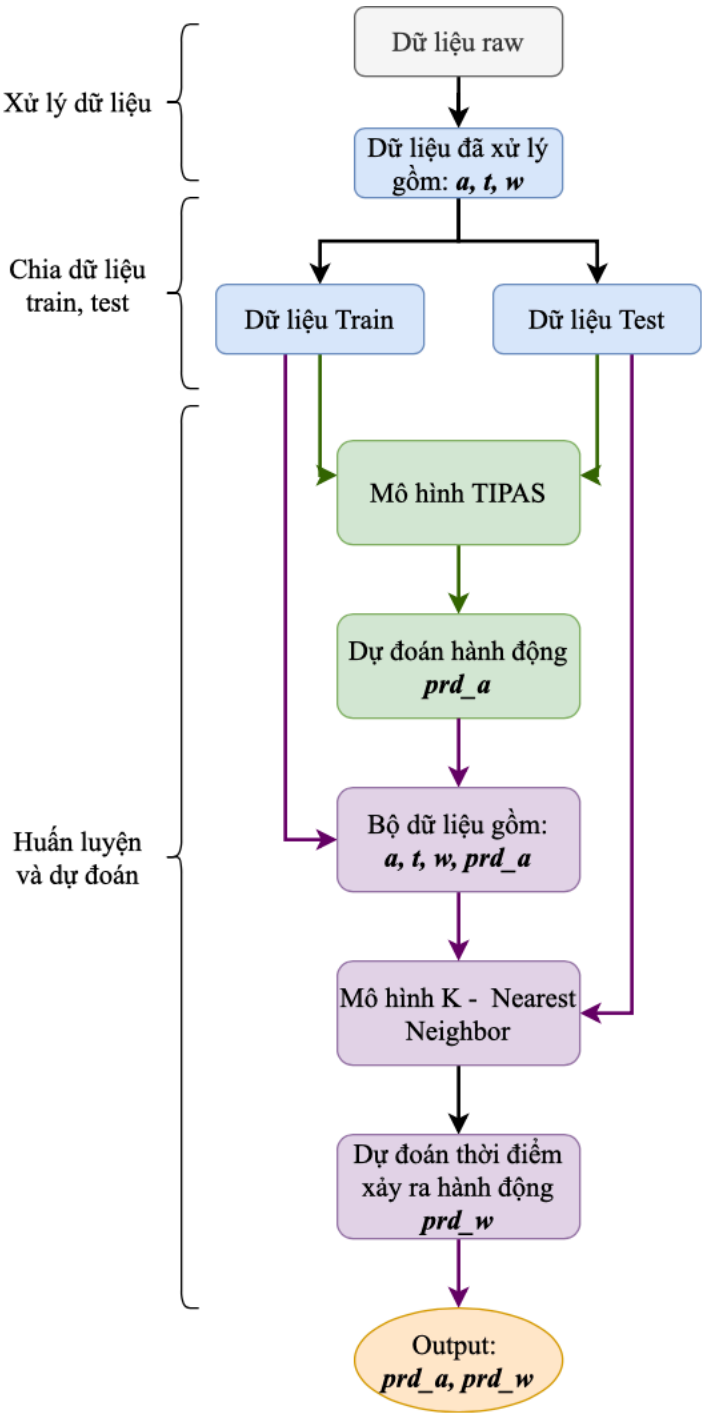
4.2.1. Mô hình tổng quát

Hầu hết các mô hình máy học hiện nay đều chỉ phù hợp để dự đoán một thành phần dữ liệu như hành động người dùng hoặc thời điểm xảy ra hành động. Thêm vào đó, các kết quả dự đoán này thường là những hành động rời rạc và không đảm bảo được tính chất của hành động như tính cá nhân hoá của hành động, hay hành động sẽ bị tác động bởi những chu kì dài hạn và ngắn hạn của mỗi người dùng. Vì thế, mô hình dự đoán của đề tài chính là bước cải tiến của những mô hình này.

Nhìn chung, mô hình dự đoán có 2 quá trình chính là: huấn luyện và dự đoán. Ở bước huấn luyện, mô hình sẽ thông qua bộ dữ liệu training để điều chỉnh các tham số sao cho phù hợp với bộ dữ liệu nhất. Cùng với đó, thông qua cách tổ chức cấu trúc dữ liệu ở bước tiền xử lý dữ liệu để đảm bảo tính chất của hành động. Cuối cùng, quá trình dự đoán sẽ được diễn ra và mang lại kết quả như mong muốn.

Tuy nhiên, để dự đoán được thời điểm xảy ra hành động thì ta cần biết hành động đó là gì? Vì thế, bài toán của đề tài được chia thành 2 bài toán con: dự đoán hành động sẽ xảy ra và dự đoán thời điểm xảy ra hành động đó.

- Đối với bài toán dự đoán hành động: sử dụng mô hình TIPAS và bộ dữ liệu gồm 3 thành phần là: ***a, t, w***.
- Đối với bài toán dự đoán thời điểm: sử dụng mô hình KNN và bộ dữ liệu gồm 4 thành phần là: ***a, t, w, prd_a***.



Hình 4-1 Framework bài toán

Mô hình dự đoán chuỗi hành động của người dùng trong tương lai gồm 2 giai đoạn chính như sau:

- Giai đoạn 1: Xử lý dữ liệu

  Ta tiến hành cấu trúc lại bộ dữ liệu theo dạng bảng để dễ truy xuất và chỉnh sửa dữ liệu. Sau đó, chọn lọc những đặc trưng cần thiết cho quá trình huấn luyện và dự đoán của mô hình.

- Giai đoạn 2: Huấn luyện và dự đoán

  Quá trình huấn luyện và dự đoán được chia thành 2 phần nhỏ là: dự đoán hành động và dự đoán thời điểm xảy ra hoạt động. Sau khi có kết quả hành động *a'* dự đoán sẽ xảy ra trong tương lai, chúng ta sử dụng kết quả này cùng với bộ dữ liệu để tiếp tục dự đoán thời điểm xảy ra hành động *a'*.

4.2.2. Phương pháp xây dựng mô hình

Bài toán được đặt ra trong Khoá luận được chia làm 2 bài toán nhỏ và chúng tôi lần lượt giải quyết các bài toán này như sau:

4.2.2.1. Bài toán dự đoán hành động sẽ xảy ra trong tương lai

a. Định nghĩa mô hình

Bằng thực nghiệm quan sát, hành động của người dùng có các tính chất là: cá nhân hoá, xảy ra theo thói quen người dùng, phụ thuộc của hành động trong chu kì ngắn hạn, có sự ảnh hưởng của chu kì dài hạn. Vì thế, chúng tôi đã mô hình hoá hành động của người dùng giống bằng quy trình điểm đa biến tạm thời (a multivariate temporal point process) cùng với cường độ time – varying [1].

Mô hình có hàm tính toán khi hành động *a* xảy ra vào thời điểm *t* trong lịch sử hành động *u* như sau:

$$\lambda_u(t,a) = \alpha_{ua} + Time_u(t,a) + ShortTerm_u(t,a) + LongTerm_u(t,a) \quad (15)$$

Trong đó:

- $\alpha_{ua} \geq 0$ thể hiện cho tính chất cá nhân hoá của hành động, vì thế có thể xem α_{ua} là một hằng số.

- $Time_u(t,a)$ thể hiện tính chất xu hướng time – varying của hành động, các sự kiện đang diễn ra đều có thể bị ảnh hưởng và tác động từ những hành động đã xảy ra trước đó. Tính chất này được mô hình hoá bằng mô hình hỗn hợp Gaussian với công thức sau:

$$Time_u(t,a) = \sum_{z \in Z} \frac{\beta_{az}}{\sqrt{2\pi\sigma^2_{az}}} exp(-\frac{(l_t - \mu_{az})^2}{2\sigma^2_{az}}) \quad (16)$$

Trong đó:

- $z \in Z$ đại diện cho biến số của mô hình hỗn hợp Gaussian và số hỗn hợp có thể được quyết định thông qua phương pháp cross – validation.
- μ_{az}, σ_{az} của mỗi hành động a và biến hỗn hợp z là giá trị trung bình và độ lệch chuẩn của phân phối Gaussian.
- $\beta_{az} \geq 0$ là giá trị quan trọng nhất của mô hình và quyết định giá trị của mô hình.
- l_t thể hiện các khung giờ trong ngày.

- $ShortTerm_u(t,a)$ thể hiện tính chất phụ thuộc lẫn nhau của hành động trong chu kì ngắn hạn. Để mô hình hoá hành động a xảy ra vào thời điểm t bị ảnh hưởng bởi hành động a' xảy ra trước đó tại thời điểm t', với $t' < t$, chúng tôi sử dụng quy trình Hawkes kết hợp với tính chất self – excitations và hàm Exponential decay. $ShortTerm_u(t,a)$ được biểu diễn qua công thức sau:

$$ShortTerm_u(t,a) = \sum_{(t',a' \in H_{ut})} \theta_{a'a} \omega_{a'a} exp(-\omega_{a'a} \Delta_{t't}) \quad (17)$$

Trong đó:

- $H_{ut} = \{(t', a') | (t', a') \in H_u, t' < t\}$ là chuỗi các sự kiện xảy ra trước thời điểm t trong quá khứ u.
- $\Delta_{t't} = t - t'$ là sự chênh lệch thời gian giữa t và t'.
- $\omega_{a'a} \geq 0$ là tham số quyết định sự ảnh hưởng của trigger hành động a' tới hành động a, đây cũng là hình dáng của phân phối mũ (shape).
- $\theta_{a'a} \geq 0$ cũng là tham số quyết định sự ảnh hưởng của trigger hành động a' tới hành động a, đây cũng là kích thước của phân phối mũ (scaling).

- $LongTerm_u(t,a)$ thể hiện tính chất hành động bị ảnh hưởng bởi chu kì dài hạn. Với tính chất này, chúng tôi sử dụng phân phối Weibull để mô hình hoá hành động. Hành động *a* xảy ra vào thời điểm *t* bị ảnh hưởng bởi hành động *a'* xảy ra trước đó tại thời điểm *t'*, với *t'* < *t* được định nghĩa như sau:

$$LongTerm_u(t,a) = \sum_{t',a' \in H_{ut}^a} \phi_{c_{t'},a} \gamma_{c_{t'},a} k_{c_{t'},a} \Delta_{t't}^{k_{c_{t'},a}-1} \exp exp\left(\gamma_{c_{t'},a} \Delta_{t't}^{k_{c_{t'},a}-1}\right) \quad (18)$$

Trong đó:

- $H_{ut} = \{(t', a') | (t', a') \in H_u, t' < t\}$ là chuỗi các sự kiện xảy ra trước thời điểm *t* trong quá khứ *u*.
- $\Delta_{t't} = t - t'$ là sự chênh lệch thời gian giữa *t* và *t'*.
- $c_{t'} \in C$ đại diện cho các khung giờ xảy ra hành động trong ngày (có 4 khung giờ: 0 – 6 giờ, 6 – 12 giờ, 12 – 18 giờ, 18 – 24 giờ).
- $\phi_{c_{t'},a} \geq 0$, $\gamma_{c_{t'},a} \geq 0$ là các trigger của hành động *a*, quyết định sự ảnh hưởng của trigger hành động *a'* tới hành động *a* tại thời điểm $c_{t'}$.
- $k_{c_{t'},a} \geq 0$ là hình dáng của phân phối Weibull (shape).

b. Huấn luyện mô hình

Để mô hình có tính hiệu quả cao hơn, chúng tôi sử dụng Maximum likelihood estimation để xác định thêm các tham biến của mô hình. Các tham số đó là $\alpha = \{\{\alpha_{ua}\}_{u \in U}\}_{a \in A}$, $\beta = \{\{\beta_{az}\}_{a \in A}\}_{z \in Z}$, $\mu = \{\{\mu_{az}\}_{a \in A}\}_{z \in Z}$, $\sigma = \{\{\sigma_{az}\}_{a \in A}\}_{z \in Z}$, $\Theta = \{\{\theta_{a'a}\}_{a \in A}\}_{a' \in A}$, $\Omega = \{\{\omega_{a'a}\}_{a \in A}\}_{a' \in A}$, $\Phi = \{\{\phi_{ca}\}_{c \in C}\}_{a \in A}$, $\Gamma = \{\{\gamma_{ca}\}_{c \in C}\}_{a \in A}$, $K = \{\{k_{ca}\}_{c \in C}\}_{a \in A}$. Tổng hợp những tham biến này gọi là **Ψ = {α, β, μ, σ, Θ, Ω, Φ, Γ, K}**.

Với chuỗi hành động của người dùng trong quá khứ $H = \{H_u\}_{u \in U}$ và diễn ra trong thời gian từ *0* tới *T*, hàm log-likelihood có thể được biểu diễn như sau:

$$L(H) = \sum_{u \in U} \sum_{n=1}^{N_u} log\lambda_u\left(t_{un}, a_{un}\right) - \sum_{u \in U} \int_0^T \sum_{a \in A} \lambda_u(t, a)dt \quad (19)$$

Từ công thức (15), (16), (17), (18), (19) chúng ta có thể viết lại hàm log-likelihood như sau:

$$L(H) = \sum_{u \in U} \sum_{n=1}^{N_u} log\{\alpha_{ua_{un}} + \sum_{z \in Z} \frac{\beta_{a_{un}z}}{\sqrt{2\pi\sigma^2_{a_{un}z}}} \exp exp\left(-\frac{\left(l_{t_{un}} - \mu_{a_{un}z}\right)^2}{2\sigma^2_{a_{un}z}}\right) + \sum_{m=1}^{n-1} \theta_{a_{um}a_{un}} \quad (20)$$

Tiếp theo, phát triển thuật toán để tìm kiếm giá trị cực đại của log-likelihood dựa trên thuật toán Expectation Maximization (EM). Đầu tiên, ở bước E – step, sử dụng 2 biến ***p, q, r*** làm trigger cho các tính chất của hành động được giới thiệu ở trên [1]. Những triggers này sẽ ước tính xác suất hành động *a* xảy ra thứ *l-th* tại thời điểm *t* của người dùng.

$$p^k_{0,un} = \frac{\alpha^k_{ua_{un}}}{R_{un}} \quad (21)$$

$$p^k_{z,un} = \frac{1}{R_{un}} \frac{\beta^k_{a_{un}z}}{\sqrt{2\pi(\sigma^k_{a_{un}z})^2}} exp(-\frac{(l_{t_{un}} - \mu^k_{a_{un}z})^2}{2(\sigma^k_{a_{un}z})^2}) \quad (22)$$

$$q^k_{um,un} = \frac{1}{R_{un}} \theta^k_{a_{um}a_{un}} \exp \omega^k_{a_{um}a_{un}} exp\left(-\omega^k_{a_{um}a_{un}} \Delta_{t_{um}t_{un}}\right) \quad (23)$$

$$r^k_{ul,un} = \{\frac{1}{R_{un}} \phi^k_{c_{ul}a_{un}} \gamma^k_{c_{ul}a_{un}} k^k_{c_{ul}a_{un}} \times \Delta^{k_{c_{ul}a_{un}}-1}_{t_{ul}t_{un}} exp(\gamma^k_{c_{ul}a_{un}} \Delta^{k^k_{c_{ul}a_{un}}}_{t_{ul}t_{un}})\} \quad (24)$$

Sau đấy, ở bước M – step sẽ lấy kết quả đã tính được ở bước E – step để gán vào giá trị các tham số của mô hình và cập nhật lại chúng sau mỗi lần tính toán. Cụ thể hơn, thông qua bất đẳng thức Jensen để tính toán lower bound của hàm log-likelihood. Sau đó, sử dụng kết quả của hàm lower bound này để lấy đạo hàm các tham số và gán giá trị cho chúng bằng 0.

$$\alpha_{ua}^{k+1} = \frac{\sum_{n=1}^{N_u} I(a_{un}=a) p_{0,un}^{k}}{T} \tag{25}$$

$$\beta_{az}^{k+1} = \frac{2T}{|U|T} \times \frac{\sum_{u\in U}\sum_{n=1}^{N_u} I(a_{un}=a) p_{z,un}^{k}}{\left(\frac{\mu_{az}^{k}}{\sqrt{2}\sigma_{az}^{k}}\right) + erf\left(\frac{T-\mu_{az}^{k}}{\sqrt{2}\sigma_{az}^{k}}\right)} \tag{26}$$

$$\theta_{a'a}^{k+1} = \frac{\sum_{u\in U}\sum_{n=1}^{N_u}\sum_{m=1}^{n-1} I(a_{um}=a', a_{un}=a) p_{um,un}^{k}}{\sum_{u\in U}\sum_{n=1}^{N_u} I(a_{un}=a')(1-exp(-\omega_{a'a}^{k}(T-t_{un})))} \tag{27}$$

$$\phi_{ca}^{k+1} = \frac{\sum_{u\in U}\sum_{n=1}^{N_u}\sum_{l=1}^{n-1} I(a_{ul}=a', a_{un}=a, c_{ul}=c) r_{ul,un}^{k}}{\sum_{u\in U}\sum_{n=1}^{N_u} I(a_{un}=a', c_{un}=c)(1-exp(-\gamma_{ca}^{k}(T-t_{un})^{r_{ca}^{k}}))} \tag{28}$$

Tuy nhiên, tham số của mô hình gồm 9 tham số là $\Psi = \{\alpha, \beta, \mu, \sigma, \Theta, \Omega, \Phi, \Gamma, K\}$ và chỉ mới có 4 tham số được sử dụng. Bởi lẽ, $\omega_{a'a}^{k+1}$, γ_{ca}^{k+1}, k_{ca}^{k+1}, μ_{az}^{k+1}, σ_{az}^{k+1} không phù hợp khi tính bằng công thức (20). Vì thế, k_{ca}^{k+1}, μ_{az}^{k+1}, σ_{az}^{k+1} sẽ được ước tính bằng giá trị lớn nhất của lower bound thông qua phương thức tối ưu hoá gradient – based cùng với phương thức Newton. Còn đối với $\omega_{a'a}^{k+1}$, γ_{ca}^{k+1} được tính qua công thức sau đây.

$$\omega_{a'a}^{k+1} = \{\sum_{u\in U}\sum_{n=1}^{N_u}\sum_{m=1}^{n-1} I(a_{um}=a', a_{un}=a) p_{um,un}^{k}\} / \{\sum_{u\in U}\sum_{n=1}^{N_u}\sum_{m=1}^{n-1} I(a_{um}=a', a_{un} \tag{29}$$

$$\gamma_{a'a}^{k+1} = \{\sum_{u\in U}\sum_{n=1}^{N_u}\sum_{m=1}^{n-1} I(a_{ul}=a, a_{un}=a, c_{ul}=c) r_{ul,un}^{k}\} / \{\sum_{u\in U}\sum_{n=1}^{N_u}\sum_{l=1}^{n-1} I(a_{ul}=a, \tag{30}$$

Kết quả của mô hình TIPAS sẽ là hành động sẽ xảy ra trong tương lai và chúng tôi sẽ sử dụng kết quả này để tiếp tục giải quyết bài toán dự đoán thời điểm xảy ra hành động trong tương lai.

4.2.2.2. Bài toán dự đoán thời điểm sẽ xảy ra hành động trong tương lai

Với bài toán con này, chúng tôi sẽ sử dụng mô hình KNN và thực hiện theo 3 bước sau đây:

Bước 1: Xử lý dữ liệu

Kết hợp bộ dữ liệu đã có sẵn và kết quả dự đoán hành động mà mô hình TIPAS đã dự đoán được, cấu trúc lại bộ dữ liệu theo dạng bảng với các thuộc tính cần thiết cho quá trình huấn luyện và dự đoán mô hình. Sau đó chọn ra các đặc trưng cần thiết là: a, t, prd_a.

Bước 2: Chia dữ liệu huấn luyện và kiểm thử sao cho cả 2 bộ dữ liệu chứa đầy đủ 295 người dùng và các thông tin về hành động của họ.

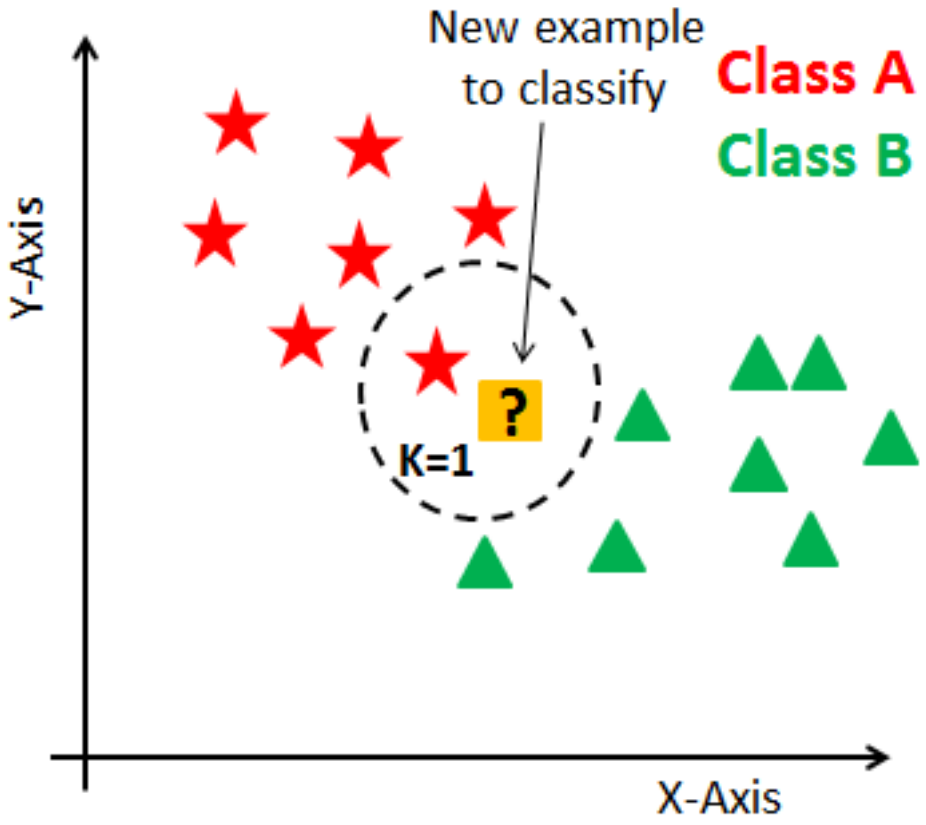
Bước 3: Huấn luyện và dự đoán mô hình

a. Cách thức hoạt động

Một câu nói thường được sử dụng đối với mô hình KNN là "Hãy cho tôi biết bạn của bạn là ai, tôi sẽ nói bạn là người như thế nào". Bởi lẽ, mô hình KNN sẽ lựa chọn phân lớp của điểm dữ liệu *x* dựa trên phân lớp của những điểm lân cận *x*. Chúng ta cũng có thể thấy cách thức hoạt động của mô hình này được phản ánh thông qua tên gọi của nó K – Nearest Neighbor (K – hàng xóm gần nhất).

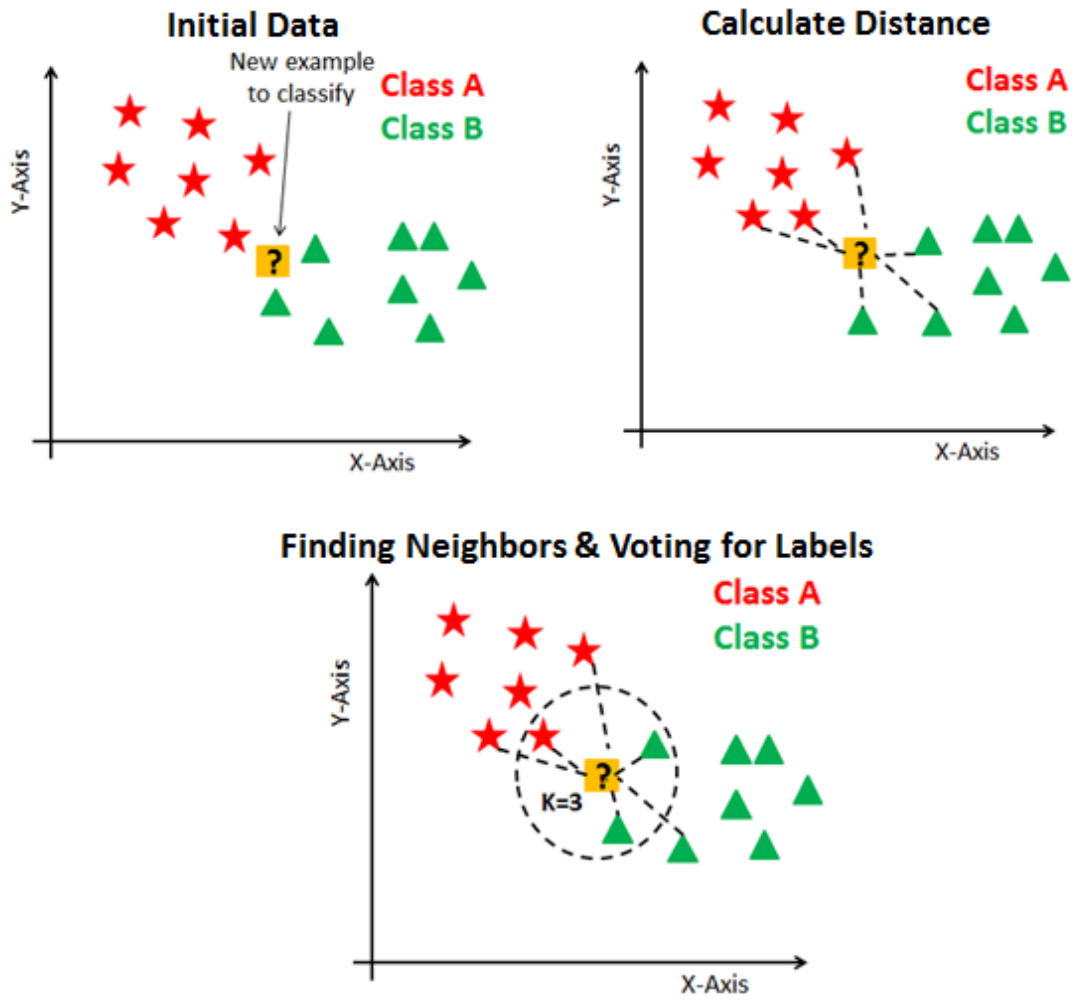
Cụ thể hơn, K là số láng giềng gần nhất hay là những điểm có vị trí gần điểm dữ liệu cần phân loại nhất. Giá trị của K sẽ là yếu tố quyết định hiệu suất dự đoán của mô hình. Khi K = 1, thì thuật toán được gọi là thuật toán láng giềng gần nhất và đây cũng là trường hợp đơn giản nhất.

Giả sử K = 1, P1 là điểm dữ liệu cần dự đoán thì đầu tiên, mô hình sẽ tìm điểm dữ liệu gần nhất với P1. Sau đó, dựa và nhãn của điểm dữ liệu này mà quyết định nhãn được gán cho P1.



Hình 4-2 Mô hình K – Nearest Neighbor tìm các điểm lân cận

Cụ thể, để xác định nhãn của điểm dữ liệu cần xác định, đầu tiên phải tìm K điểm gần nhất với P1 và sau đó xét các nhãn của K điểm này. Tiếp theo, đếm số lượng điểm dữ liệu thuộc về từng nhãn, lớp có nhiều điểm K lân cận thuộc về nhất sẽ là lớp của điểm dữ liệu cần phân loại.



Hình 4-3 Cách thức hoạt động của mô hình K – Nearest Neighbor

Thuật toán của KNN xác định lớp cho mẫu E [30]:

- Tính khoảng cách giữa E và tất cả các mẫu trong tập huấn luyện.
- Chọn K mẫu gần nhất với E trong tập huấn luyện.
- Gán E vào lớp có nhiều mẫu nhất trong số K mẫu láng giềng đó (hoặc E nhận giá trị trung bình của K mẫu)

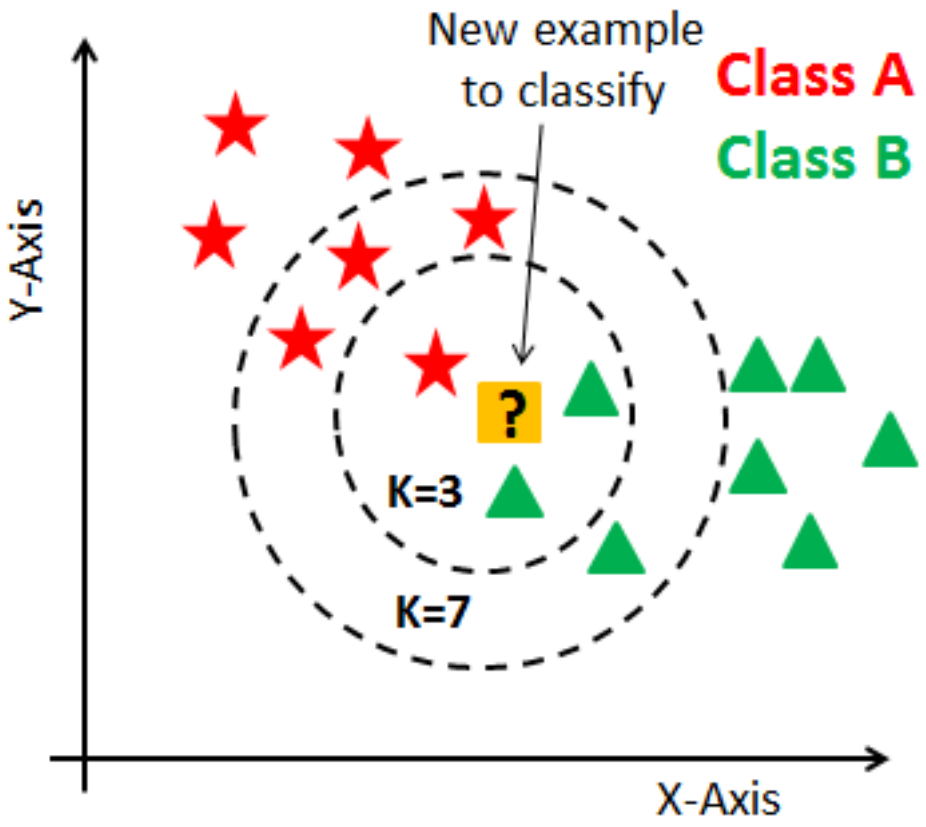
Trong đó, khoảng cách giữa các điểm có thể được tính bằng cách sử dụng các thước đo khoảng cách như khoảng cách Euclidean, Hamming, Manhattan và Minkowski. Ví dụ, khoảng cách Euclidean giữa $X = (x_1, ..., x_n)$ và $Y=(y_1, ..., y_n)$ là:

$$D(X,Y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \quad (31)$$

b. Phương pháp chọn giá trị K – Neighbor cho mô hình

Thực chất không có phương pháp chính xác để xác định giá trị K của mô hình KNN. Bởi vì, nhiều nghiên cứu đã chỉ ra rằng không có giá trị K cụ thể nào là phù hợp với tất cả các loại dữ liệu có kích thước và tính chất khác nhau [31]. Nếu ta chọn giá trị K nhỏ thì sẽ ảnh hưởng đến kết quả dự đoán của mô hình, còn nếu ta chọn giá trí K quá lớn thì sẽ rất tốn kém chi phí tính toán và bộ nhớ lưu trữ. Nhiều nghiên cứu cũng đã chỉ ra rằng, khi chọn giá trị K nhỏ thì sẽ có độ linh hoạt và có độ chệch thấp nhưng phương sai cao và một số lượng lớn hàng xóm sẽ có ranh giới quyết định mượt mà hơn, nghĩa là phương sai thấp hơn nhưng độ chệch cao hơn.

Vì thế, chúng tôi đã xác định giá trị này theo phương thức chọn là số lẻ nếu số nhãn là số chẵn và ngược lại. Thêm vào đó, bằng cách tạo mô hình trên các giá trị khác nhau của K và kiểm tra hiệu suất của chúng, chúng tôi đã chọn ra giá trị K phù hợp khi dự đoán thời điểm xảy ra hành động là k = 115 cho số class = 4.



Hình 4-4 Ví dụ chọn giá trị K lân cận

4.3. Kết luận

Ở chương 4, chúng tôi đã giới thiệu tổng quan về mô hình bài toán và các cơ sở hình thành nên mô hình dự đoán hành động và thời điểm xảy ra hành động đó. Khoá luận đã trình bày chi tiết những nội dung sau:

- Xây dựng tập dữ liệu huấn luyện và kiểm thử.
- Lựa chọn các đặc trưng của bộ dữ liệu phù hợp với mô hình dự đoán.
- Giải thích mô hình tổng quát và phương pháp xây dựng mô hình.

Với những nền tảng khoa học trên, mô hình dự đoán chuỗi hành động của con người trong thế giới thực do Khoá luận đề xuất đã giúp giảm thời gian dự đoán và tăng tỷ lệ dự đoán chính xác.

Chương 5. KẾT QUẢ THỰC NGHIỆM VÀ ĐÁNH GIÁ

5.1. Dữ liệu thực nghiệm

Đề tài thực nghiệm trên bộ dữ liệu hành động của người dùng trong thế giới thực gồm 234.665 hành động được thực hiện bởi 295 người trong 17 tháng.

Mô tả dữ liệu:

Cụ thể hơn, bộ dữ liệu gồm 5 thành phần chính như sau:

- ***a***: (*action*) id hành động người dùng
- ***t:*** *(time)* thời gian hành động ***a*** kéo dài, được tính bắt đầu từ thời điểm quan sát thực nghiệm cho tới thời điểm hành động được ghi lại và có đơn vị là phút.
- ***w***: (*window of time*) thời điểm diễn ra hành động ***a***
- ***prd_a***: id hành động tiếp theo của người dùng sau hành động ***a***
- ***prd_w*:** thời điểm diễn ra hành động ***prd_a***

Trong đó:

- **action_id**: các id của 10 nhãn hành động người dùng. Được sắp xếp từ 0 đến 9 lần lượt như sau:

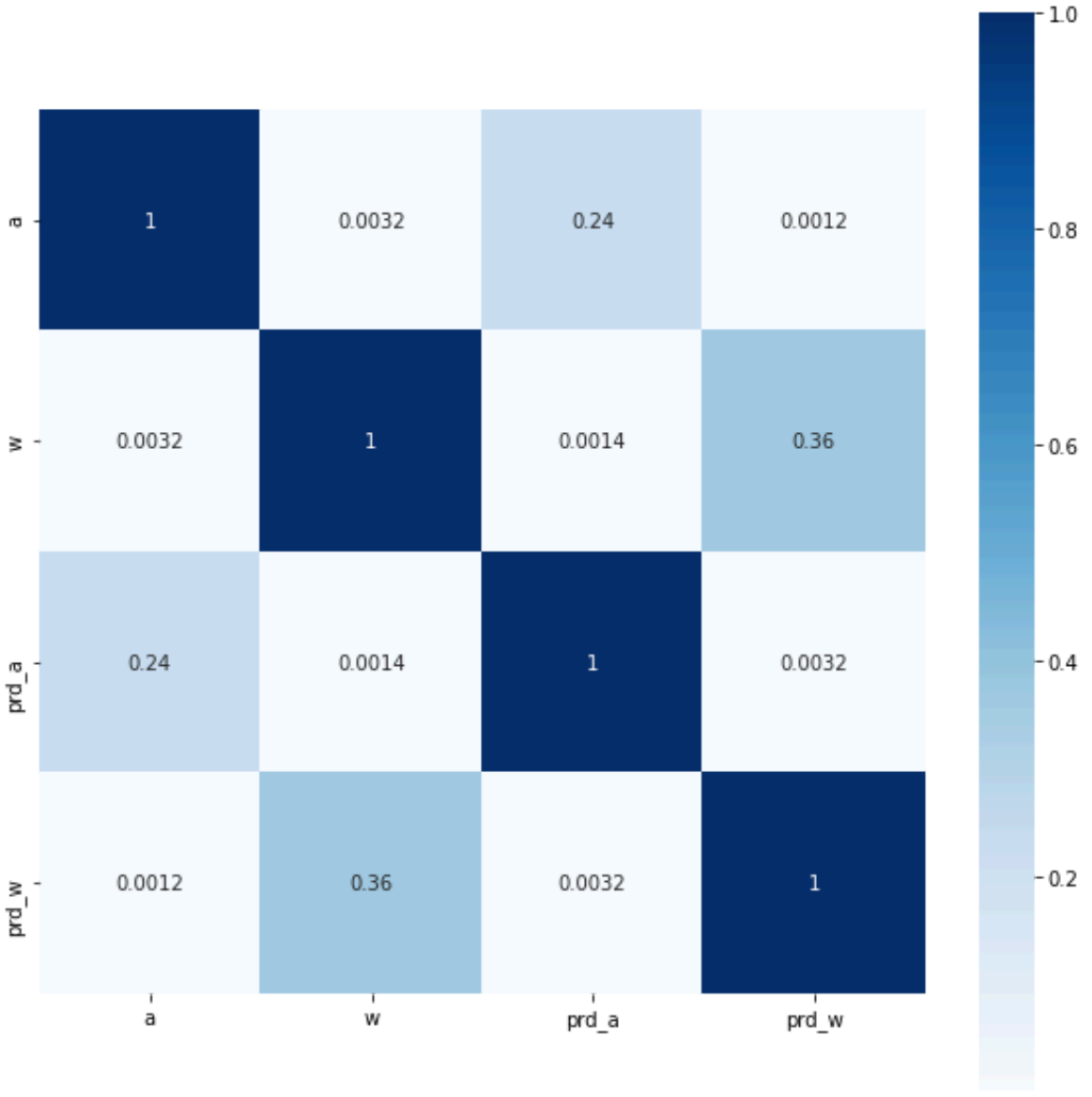
  0: Uống
  1: Ngủ
  2: Nhịp tim
  3: Chạy bộ
  4: Cân nặng
  5: Ăn
  6: Đi bộ
  7: Đạp xe
  8: Tập thể dục
  9: Giãn cơ

- ***w***: thời điểm xảy ra hành động, được chuẩn hoá 4 khung thời gian được gắn với id từ 0 đến 3 lần lượt như sau:

  0: 0 – 6 giờ
  1: 6 – 12 giờ
  2: 12 – 18 giờ
  3: 18 – 24 giờ

5.1.1. Độ phức tạp của dữ liệu

Dựa vào ma trận hỗn loạn sau đây, chúng ta có thể thấy rằng bộ dữ liệu có tính phức tạp rất cao và độ tương quan giữa các thành phần dữ liệu hầu như rất thấp. Tuy nhiên, các cặp hành động [*a, prd_a*], [*w, prd_w*] có mối tương quan rõ rệt hơn với những thành phần khác. Từ đây, có thể thấy rằng các thông số của hành động *a* và hành động xảy ra tiếp theo *prd_a* có sự ảnh hưởng lẫn nhau.



Hình 5-1 Ma trận hỗn loạn của bộ dữ liệu

Bên cạnh đó, với tổng số 295 người dùng, số lượng hành động của mỗi người dùng là khác nhau. Người dùng có số lượng hành động ít nhất là 6 hành động, người dùng có số lượng hành động nhiều nhất là 2128 hành động và trung bình mỗi người có 796 hành động.

Vì thế, thông qua tính chất này của hành động, mô hình có khả năng dự đoán hành động mà vẫn đảm bảo tính chất cá nhân hoá và bị ảnh hưởng bởi các chu kỳ ngắn hạn và dài hạn.

Bảng thống kê số hành động của người dùng như sau:

Id người dùng	Số lượng hành động
6	6
13	25
15	29
19	19
20	77
21	61
22	22
23	155
24	162
25	49
26	51
27	26
28	109
29	197
30	87
31	240
32	249
33	129
34	264
35	206
36	316
37	181
38	74
39	267
40	313
41	321
42	534
43	504
44	430
45	573
46	585
47	414
48	659
49	289
50	344
51	601
52	611
53	677
54	901
55	593
56	713
57	449
58	513
Id người dùng	**Số lượng hành động**
59	523
60	531
61	481
62	977
63	1241
64	1136
65	831
66	847
67	1122
68	1338
69	886
70	1035
71	980
72	1421
73	1370
74	1023
75	1035
76	1425
77	685
78	694
79	1638
80	1658
81	1361
82	973
83	1395
84	1244
85	1259
86	1701
87	1632
88	1131
89	1057
90	1778
91	1351
92	1274
93	1102
94	1024
95	1223
96	1710
97	1249
98	1068
99	2058
100	1880
101	1201
Id người dùng	**Số lượng hành động**
102	1111
103	1021
104	1648
105	1248
106	841
107	1379
108	1819
109	1944
110	1200
111	1211
112	1110
113	2128
114	1130
115	1255
116	1265
117	1624
118	1053
119	1888
120	1310
121	1321
122	838
123	1707
124	1723
125	1364
126	1499
127	631
128	1525
129	1536
130	775
131	1951
132	1180
133	791
134	799
135	1609
136	1350
137	1905
138	1368
139	1517
140	1252
141	840
142	1692
143	993
144	1144
Id người dùng	**Số lượng hành động**
145	577
146	1742
147	1460
148	1471
149	593
150	1192
151	900
152	1359
153	1064
154	1683
155	925
156	621
157	1561
158	942
159	1581
160	1432
161	641
162	1931
163	973
164	653
165	985
166	1486
167	665
168	1335
169	1344
170	846
171	1700
172	685
173	1721
174	1039
175	1567
176	875
177	880
178	1238
179	891
180	359
181	901
182	363

183	1093
184	1099
185	737
186	1296
187	373
188	936
189	565
190	567
191	951
192	574
193	577
194	1159
195	777
196	586
197	197
198	984
199	1584
200	1195
201	401
202	1006
203	203
204	204
205	817
206	1231
207	207
208	1243
209	625
210	1046
211	1051
212	634
213	1697
214	854
215	643

216	216
217	217
218	435
219	219
220	220
221	661
222	443
223	445
224	670
225	897
226	451
227	227
228	455
229	1141
230	1375
231	691
232	232
233	696
234	467
235	469
236	236
237	237
238	238
241	241
242	242
243	485
244	487
245	245
246	491
247	493
248	496
249	745
250	250

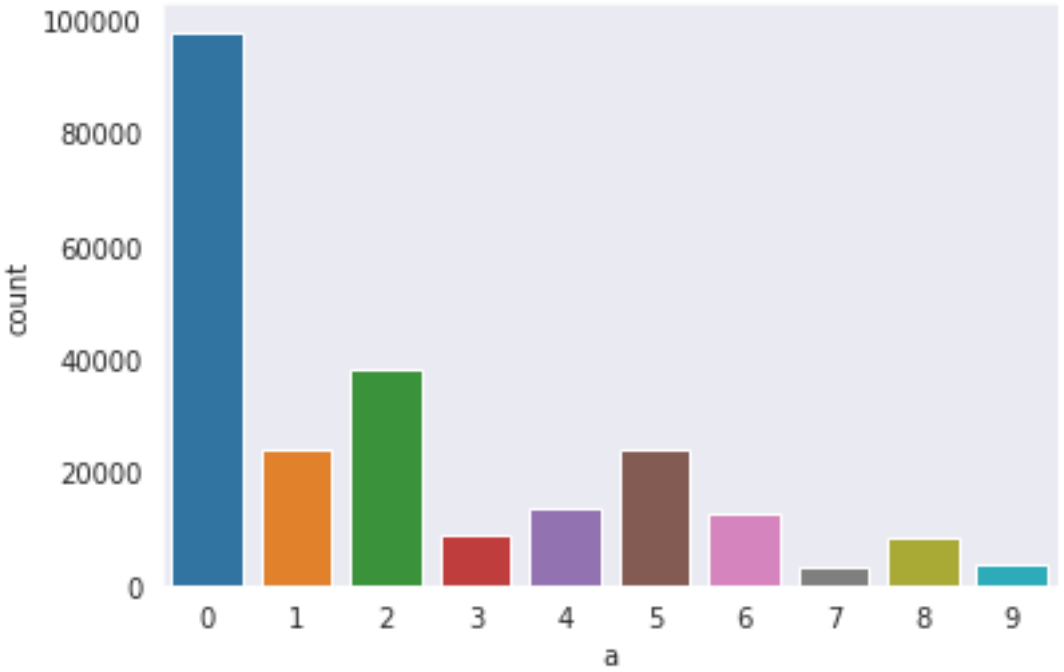
251	751
252	1002
253	757
254	1012
255	763
257	257
258	515
260	519
261	261
262	262
263	525
267	267
268	268
269	564
270	270
271	270
272	272
273	1089
274	274
276	276
281	281
283	283
284	284
285	853
286	856
288	287
291	291
292	291
294	294
296	591
299	299
301	301
302	302

303	605
304	304
308	308
309	309
311	311
317	317
323	323
325	325
326	326
331	331
332	332
339	677
340	340
344	344
348	348
359	717
360	1078
392	392
396	395
402	402
403	403
405	405
415	415
421	421
428	428
442	442
467	466
606	606
629	629

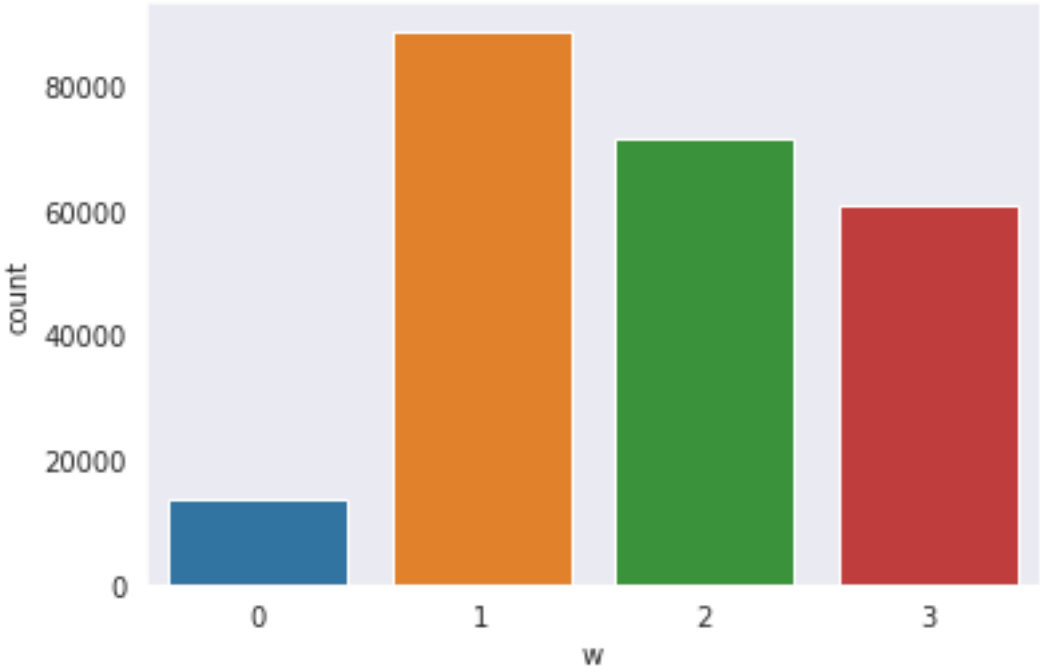
Bảng 5-1 Bảng thống kê số hành động của người dùng

5.1.2. Cấu trúc phân bố dữ liệu

Mục tiêu của đề tài là dự đoán hành động tiếp theo và thời điểm xảy ra hành động đó, vậy nên chúng tôi tập trung vào khảo sát tần số xuất hiện của các hành động và thời điểm.



Hình 5-2 Thống kê tần số xuất hiện của các hành động



Hình 5-3 Thống kê tần số xuất hiện của các thời điểm xảy ra hành động

Có thể thấy, bộ dữ liệu được sử dụng là bộ dữ liệu không cân bằng. Con người có xu hướng thực hiện hành động có id là 0 (hành động Uống) rất nhiều, nhưng các hành động có id 7, 8, 9 là đạp xe, tập thể dục, giãn cơ thì lại rất ít. Tương tự như vậy, con người thường hoạt động vào khung giờ từ 6 giờ sáng tới 12 giờ trưa và rất ít hoạt động vào khoảng 0 giờ tới 6 giờ sáng.

5.2. Độ đo đánh giá

5.2.1. Accuracy Score

Accuracy Score là độ đo ước tính khả năng dự đoán của mô hình dự đoán phân lớp. Độ đo này sẽ sử tính tỷ số kết quả dự đoán chính xác trên kết quả dự đoán đã có sẵn của bộ dữ liệu.

Công thức Accuracy Score như sau:

$$Accuracy\ Score\ = \frac{TP+TN}{TP+TN+FP+FN} \quad (32)$$

Trong đó:

- TP: True Positives

- TN: True Negatives
- FP: False Positives
- FN: False Negatives

Đối với bài toán phân lớp nhiều lớp, độ đo Accuracy Score sẽ tính tổng TP của các class chia cho tổng TP và FP các class

Ví dụ:

Các class thực

Các class dự đoán	A	B	C	D
A	50	3	0	0
B	26	8	0	1
C	20	2	4	0
D	12	0	0	1

Bảng 5-2 Ví dụ về độ đo Accuracy Score

$$Accuracy\ Score = \frac{TP(class\ A)+TP\ (class\ B)+TP(class\ C)+TP(class\ D)}{TP(class\ A)+FP(class\ A)+TP(class\ B)+FP(class\ B)+TP(class\ C)+FP(class\ C)+TP(class\ D)+FP(class\ D)}$$

Vậy độ đo Accuracy Score của ví dụ trên là 49,6%.

5.2.2. Độ đo Macro Average Recall

Macro Average Recall là trung bình cộng của các Recall theo class và Recall được định nghĩa là tỉ lệ số điểm true positive trong số những điểm thực sự là positive (TP + FN) [32]. Vì vậy, có thể nói Macro Average Recall là độ đo đánh giá tính hiệu quả của các bài toán phân lớp.

Ngoài ra, Recall cao đồng nghĩa với việc True Positive Rate cao, nghĩa là tỉ lệ bỏ sót các điểm thực sự positive là thấp. Khi Recall = 1, tức là mọi điểm positive đều được tìm thấy, điều này nghĩa là mô hình dự đoán chính xác 100% các nhãn positive. Vì thế giá trị độ đo Recall càng gần về phía 1 thì càng tốt. Do đó, tỷ số Recall càng cao thì tỷ lệ dự đoán chính xác càng cao. Điều này kéo theo giá trị độ đo Macro Average Recall càng cao càng tốt.

Công thức của Recall:

$$Recall = \frac{TP}{TP+FN} \quad (33)$$

Công thức Macro Average Recall chính là trung bình Recall của các class:

$$Macro\ Average\ Recall = \frac{1}{n} \sum_{class=1}^{n} Recall_{class} \quad (34)$$

Trong đó:

- TP: True Positives
- TN: True Negatives
- FP: False Positives
- FN: False Negatives
- n: số class

5.3. Phương pháp thực nghiệm

5.3.1. Các phương pháp so sánh

Để đảm bảo mô hình được sử dụng trong đề tài là mô hình có hiệu suất dự đoán tốt nhất, chúng tôi sử dụng mô hình Gaussian Naive Bayes, Logistic Regression, Support Vector Classifier, K – Nearest Neighbor và TIPAS thực nghiệm 14 trường hợp sau đây.

Phương pháp thực nghiệm	Đặc trưng				Dự đoán	
	a	t	w	prd_a	prd_a	prd_w
Gaussian NB	X	X	X		X	
Logistic Regression	X	X	X		X	
SVC	X	X	X		X	
KNN	X	X	X		X	
TIPAS	X	X	X		X	
Gaussian NB	X		X	X		X

Logistic Regression	X		X	X		X
SVC	X		X	X		X
KNN	X		X	X		X
Gaussian NB	X	X	X	X		X
Logistic Regression	X	X	X	X		X
SVC	X	X	X	X		X
TIPAS	X	X	X	X		X
KNN	X	X	X	X		X

Bảng 5-3 Các phương pháp so sánh

Trong đó:

- Đối với bài toán dự đoán hành động sẽ xảy ra trong tương lai, chúng tôi sử dụng 3 đặc trưng là (*a, t, w*) để dự đoán ra hành động *prd_a*.
- Đối với bài toán dự đoán thời điểm xảy ra hành động trong tương lai, chúng tôi xét 2 trường hợp sau:
    1. Sử dụng 3 đặc trưng: *a, w, prd_a* với *prd_a* là kết quả dự đoán hành động của bài toán dự đoán hành động.
    2. Sử dụng 4 đặc trưng: *a, t, w, prd_a* với *prd_a* là kết quả dự đoán hành động của bài toán dự đoán hành động.

Phương pháp được sử dụng trong đề tài là:

- TIPAS với 3 đặc trưng (*a, t, w)* khi dự đoán hành động.
- KNN với 4 đặc trưng *(a, t, w, prd_a)* khi dự đoán thời điểm.

5.3.2. Xử lý dữ liệu

Đối với bài toán dự đoán thời điểm xảy ra hành động, để thuận tiện cho việc truy xuất thông tin và xử lý thông tin, chúng tôi cấu trúc lại bộ dữ liệu theo dạng bảng và chọn các đặc trưng quan trọng.

1000 10 40320 4

100 4:504.65:1 2:505.94:1 2:507.28:1 2:509.37:1 6:678.03:1 1:1011.64:2 6:1114.24:3 2:1205.46:3 1:1319.94:3 1:2008.15:1 2:2096.67:1 2:3217.66:0 1:3330.89:1 6:3435.01:1 6:4075.85:3 1:4683.93:1 6:5275.19:2 6:6628.69:2 2:6648.90:2 6:9505.34:2

198 1:450.33:1 5:629.00:1 4:925.32:2 5:1025.19:2 4:1077.10:2 1:1960.74:1 4:1987.13:1 5:3282.43:1 4:3363.13:1 5:3433.00:1 5:3605.43:2 4:3714.73:2 5:4199.80:3 5:5194.07:2 5:5395.97:2 5:5396.37:2 5:5397.48:2

…

Dữ liệu ban đầu

⇩

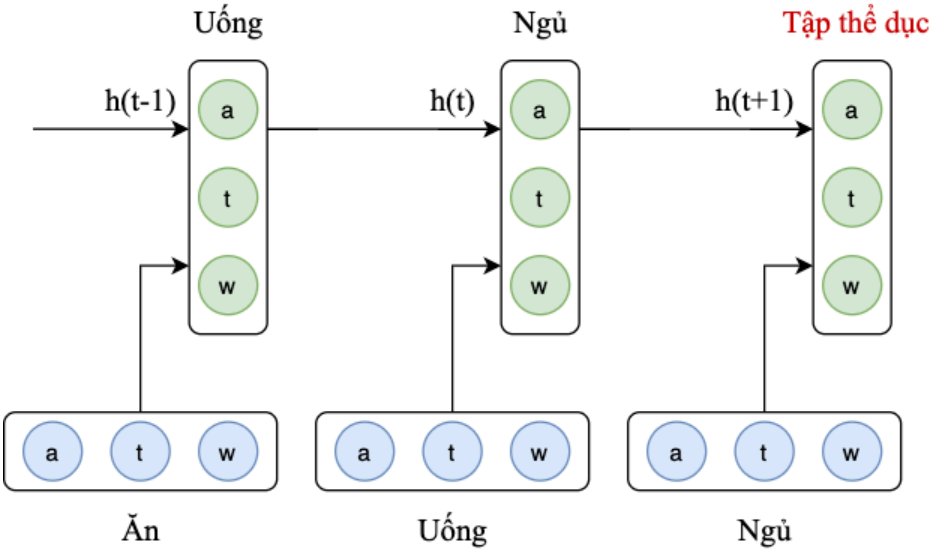
user	a	t	w	prd_a	prd_w
6	1	7734.2	1	2	1
6	2	7734.63	1	0	1
6	0	7738.51	1	1	1
6	1	12085.47	1	1	1
13	2	449.33	1	1	1
13	1	4833.99	1	4	1
13	4	4867.23	1	4	1
100	0	5219.02	2	0	3
100	0	5606.45	3	0	0
100	0	6076.91	0	0	1
100	0	6316.37	1	1	1
101	0	557	1	2	1
101	2	557.2	1	0	1
101	0	1895.86	1	3	2
101	3	2209.42	2	0	3
…	…	…	…	…	…

Dữ liệu sau khi cấu trúc lại

Sau khi cấu trúc lại dữ liệu, chúng ta có thể dễ dàng nhận thấy rằng, hành động *prd_a* sẽ được dự đoán dựa trên thông tin hành động *a*.

Ví dụ: tại thời điểm *t*, người dùng đang thực hiện hành động "Ngủ" và chúng ta đã biết được rằng trong quá khứ, người này có lịch sử hoạt động *h(t)* và đã "Ăn" sau đó "Uống".

Vậy, với thông tin dữ liệu đầu vào như thế, yêu cầu của bài toán là phải dự đoán ra hành động "Tập thể dục" sau khi người đó thức dậy.



Hình 5-4 Ví dụ hành động người dùng

Trong đó:

- h là lịch sử hoạt động của người dùng (history)
- a là hoạt động của người dùng (action)
- t là thời gian diễn ra hành động (times)
- w là thời điểm xảy ra hành động (window of times)
- t là thời điểm hiện tại

Sau đó, chia bộ dữ liệu thành 2 bộ dữ liệu train và test để huấn luyện và kiểm thử mô hình. Đặc biệt, 2 bộ dữ liệu này phải đảm bảo có đầy đủ 295 người dùng và số lượng hành động của mỗi người dùng trong 2 bộ dữ liệu này được chúng tôi chia theo tỷ lệ 80:20 (80% train và 20% test).

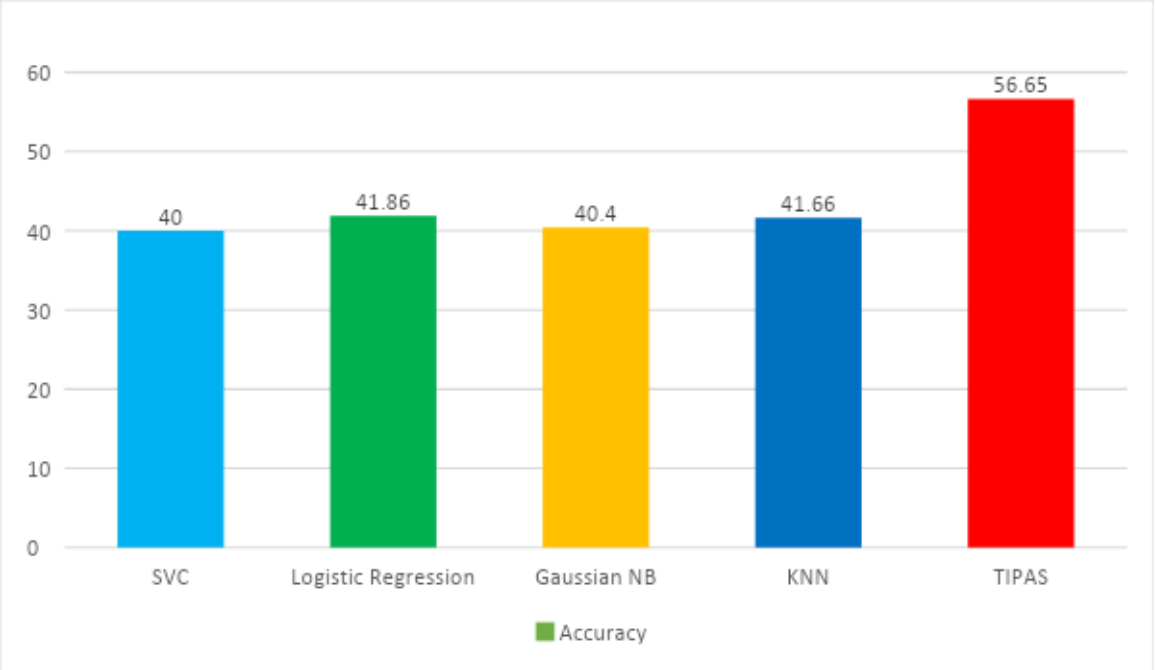
5.3.3. Huấn luyện mô hình

Để đảm bảo tính cạnh tranh công bằng và chứng minh hiệu suất dự đoán của mô hình được sử dụng trong đề tài so với các phương pháp baseline, tất cả thí nghiệm của chúng tôi được thực hiện trên máy tính hệ điều hành Mac OS, Ram 8GB.

5.4. Kết quả thực nghiệm

Sau khi cài đặt thực nghiệm, mô hình của đề tài đã dự đoán thành công hành động của người dùng trong thế giới thực mà vẫn đảm bảo được các tính chất của hành động (cá nhân hoá, xảy ra theo xu hướng, bị tác động bởi các chu kỳ). Số liệu thống kê cụ thể sau đây:

So sánh độ đo Accuracy khi dự đoán hành động sẽ xảy ra tiếp theo

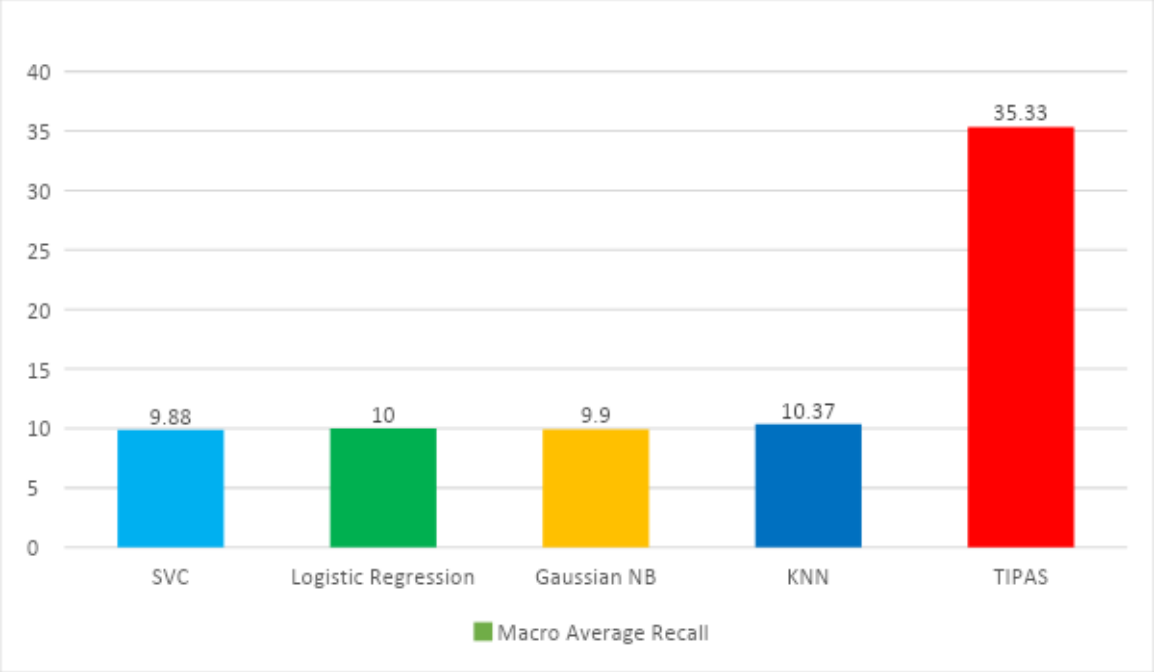


Hình 5-5 Kết quả tỷ lệ dự đoán chính xác khi dự đoán hành độngbằng độ đo Accuracy

Mô hình	Accuracy
SVC	40%
Logistic Regression	41,86%
Gaussian Naive Bayes	40,4%
KNN	41,66%
TIPAS	56,65%

Bảng 5-4 Kết quả tỷ lệ dự đoán chính xác khi dự đoán hành động bằng độ đo Accuracy

So sánh độ đo Macro Average Recall khi dự đoán hành động xảy ra tiếp theo

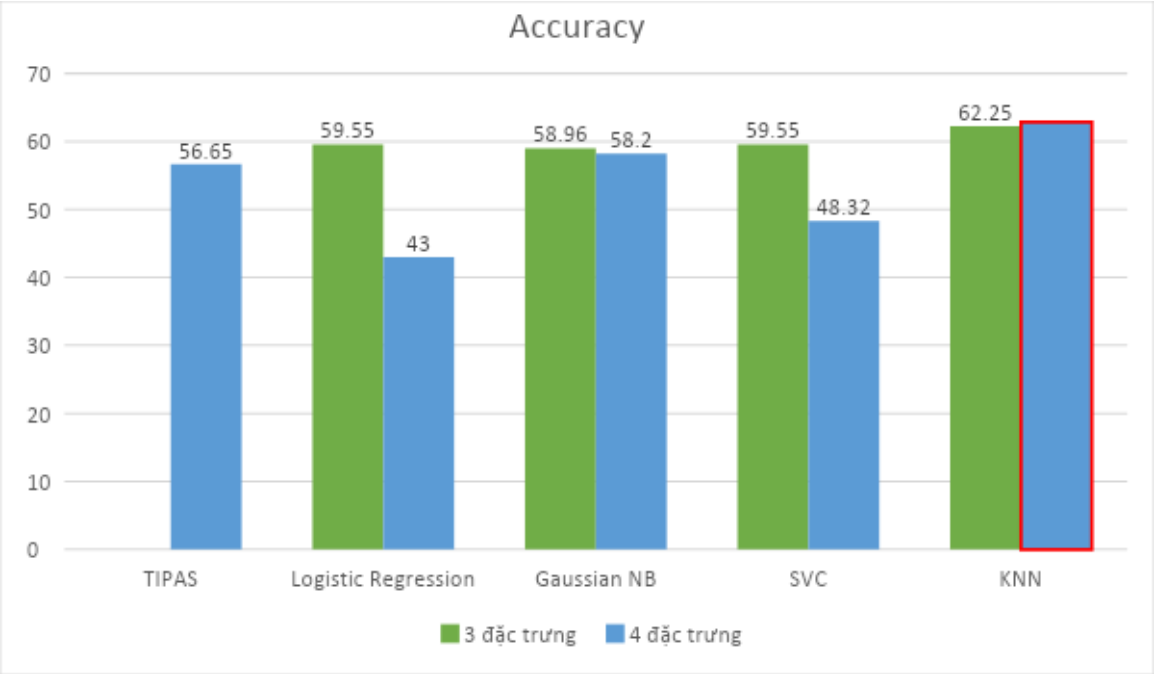


Hình 5-6 Kết quả tỷ lệ dự đoán chính xác khi dự đoán hành động bằng độ đo Macro Average Recall

Mô hình	Macro Average Recall
SVC	9,88%
Logistic Regression	10%
Gaussian Naive Bayes	9,9%
KNN	10,36%
TIPAS	35,33%

Bảng 5-5 Kết quả tỷ lệ dự đoán chính xác khi dự đoán hành động bằng độ đo Macro Average Recall

So sánh độ đo Accuracy khi dự đoán thời điểm xảy ra hành động

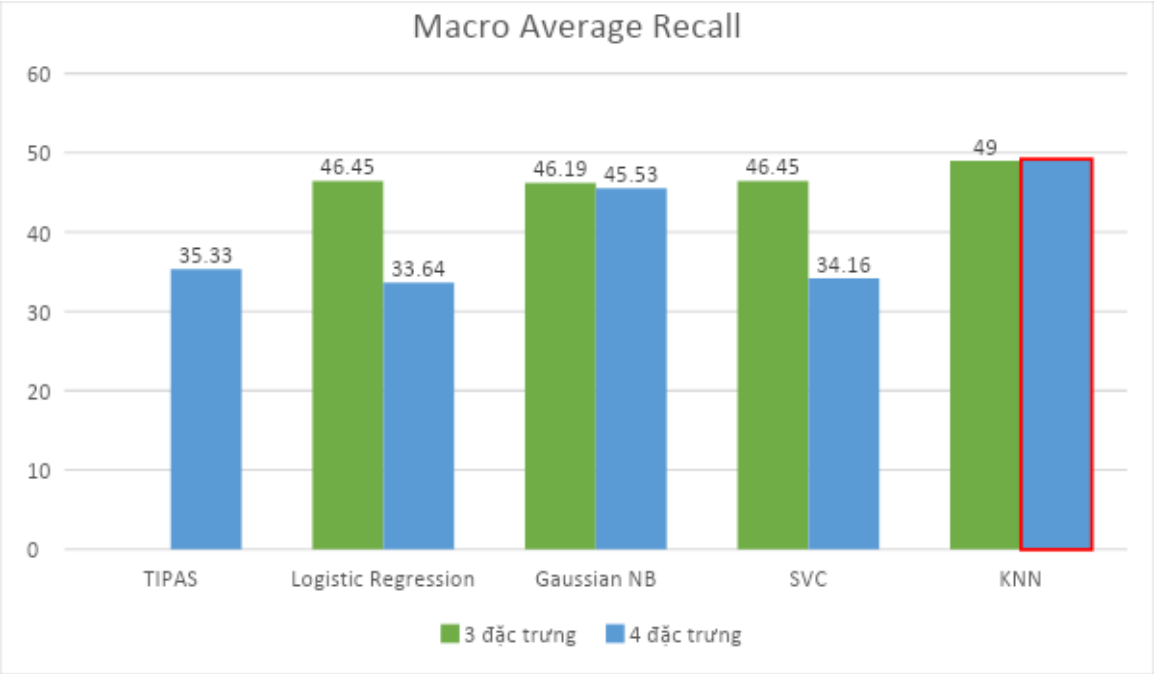


Hình 5-7: Kết quả tỷ lệ dự đoán chính xác khi dự đoán thời điểm xảy ra hành động bằng độ đo Accuracy

Mô hình	3 đặc trưng	4 đặc trưng
TIPAS	-	56,65%
Logistic Regression	59,55%	43%
Gaussian Naive Bayes	58,96%	58,2%
Support Vector Classifier	59,55%	48,32%
K – Nearest Neighbor	62,25%	62,82%

Bảng 5-6 *Kết quả tỷ lệ dự đoán chính xác khi dự đoán thời điểm xảy ra hành động bằng độ đo Accuracy*

So sánh độ đo Macro Average Recall khi dự đoán thời điểm xảy ra hành động



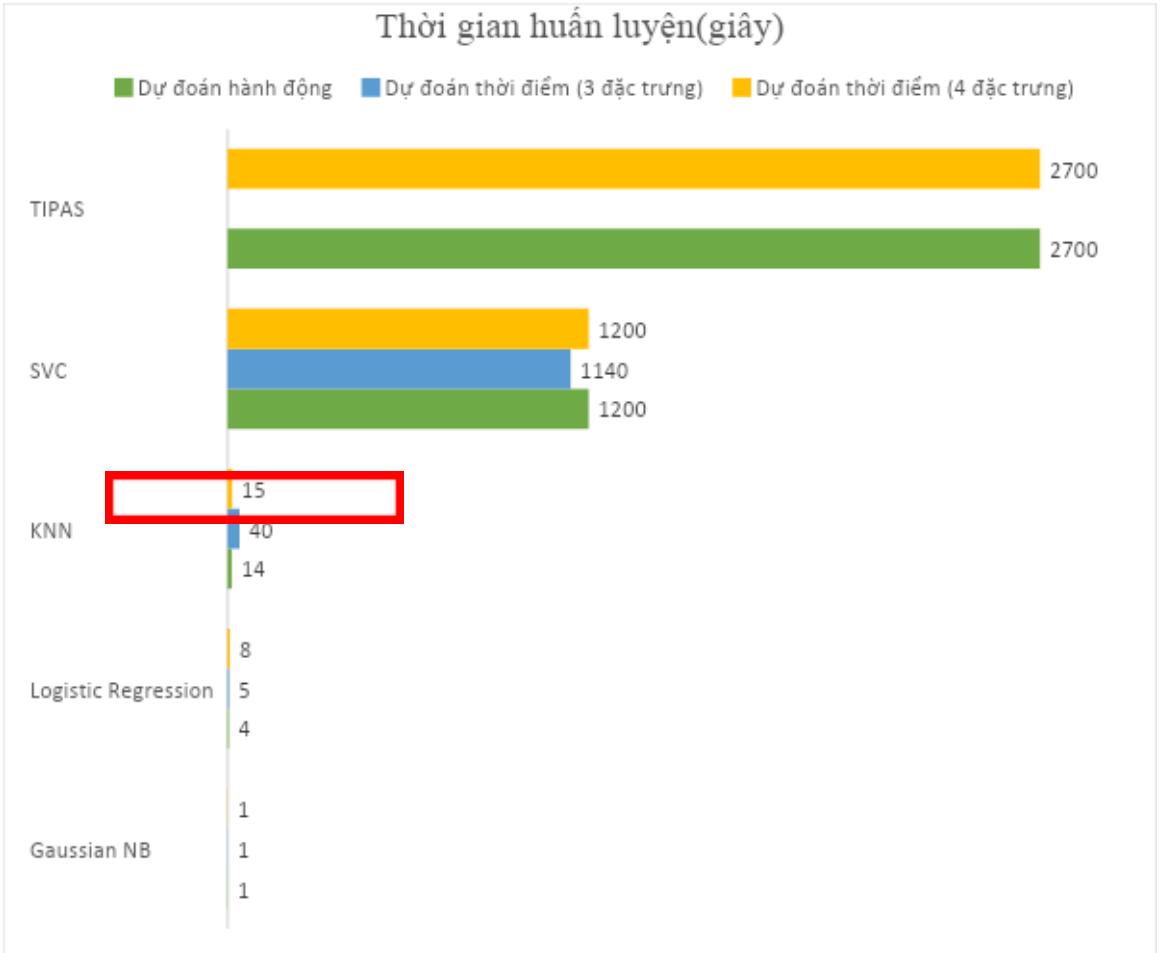
Hình 5-8 Kết quả tỷ lệ dự đoán chính xác khi dự đoán thời điểm xảy ra hành động bằng độ đo Macro Average Recall

Mô hình	3 đặc trưng	4 đặc trưng
TIPAS	-	35,33%
Logistic Regression	46,45%	33,64%
Gaussian Naive Bayes	46,19%	45,53%
Support Vector Classifier	46,45%	34,45%
K – Nearest Neighbor	49%	49,2%

Bảng 5-7 Kết quả tỷ lệ dự đoán chính xác khi dự đoán thời điểm xảy ra hành động bằng độ đo Macro Average Recall

So sánh thời gian

Mỗi phương pháp thực nghiệm sẽ có tính chất riêng và phương thức huấn luyện mô hình riêng. Ví dụ, mô hình Support Vector Classifier hay Gaussian Naive Bayes, TIPAS sẽ tính xác suất từng điểm dữ liệu rơi vào từng class và chọn ra class có xác suất cao nhất, mô hình Logistic Regression sẽ tìm phương trình hồi quy phù hợp với bộ dữ liệu nhất, còn đối với mô hình K – Nearest Neighbor chỉ cần xét class của các điểm dữ liệu trong vùng K – Neighbor lân cận để quyết định nhãn gán cho điểm dữ liệu. Do đó, thời gian huấn luyện của từng mô hình này sẽ có sự chênh lệch đáng kể. Mặc dù vậy, tuy nhiên mục tiêu của đề tài là chọn ra mô hình máy học có kết quả dự đoán cao và thời gian huấn luyện nhanh chóng để tiết kiệm chi phí tối đa. Vì thế, chúng tôi so sánh thời gian huấn của từng phương pháp baseline so với mô hình như sau.



Hình 5-9 Thời gian huấn luyện mô hình(giây)

	Dự đoán hành động	Dự đoán thời điểm (3 đặc trưng)	Dự đoán thời điểm (4 đặc trưng)
TIPAS	2700	-	2700
SVC	1200	1140	1200
Logistic Regression	4	5	8
Gaussian NB	1	1	1
KNN	14	40	15

Bảng 5-8 Thời gian huấn luyện mô hình chuỗi hành động (giây)

Tóm lại, với độ đo *Accuracy*, tỷ lệ dự đoán chính xác của mô hình đối với 2 output là:

- Hành động xảy ra tiếp theo *a'*: 56,65% trong thời gian 45 phút.
- Thời điểm xảy ra hành động *a'*: 62,82% trong thời gian 15 giây.

Bên cạnh đó, với công việc dự đoán thời điểm xảy ra hành động, mô hình được sử dụng trong đề tài khi sử dụng độ đo *Macro Average Recall* để đánh giá thì có độ chính xác cao hơn các phương pháp baseline khoảng từ 3% đến 20%.

Chương 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

6.1. Kết quả đạt được

Thông qua độ đo Accuracy và Macro Average Recall, tỷ lệ dự đoán chính xác hành động xảy ra tiếp theo của mô hình TIPAS lớn hơn các phương pháp baseline khoảng 20%, tỷ lệ dự đoán thời điểm xảy ra hành động của mô hình K – Nearest Neighbor cao hơn các phương pháp baseline từ 3% tới 14%.

Bên cạnh đó, thời gian huấn luyện và dự đoán của mô hình TIPAS khá cao (khoảng 45 phút), còn đối với mô hình KNN, thời gian dự đoán khá nhanh (khoảng 15 giây). Mặc dù thời gian huấn luyện và dự đoán của 2 mô hình được sử dụng trong đề tài còn chậm hơn các phương pháp baseline, thế nhưng tỷ lệ dự đoán của 2 mô hình này thì lại cao hơn hẳn những phương pháp khác. Hơn nữa, khoảng thời gian này cũng là lượng thời gian có thể chấp nhận được. Vì thế, chúng tôi đã sử dụng mô hình TIPAS để huấn luyện và dự đoán hành động sẽ xảy ra, mô hình KNN để huấn luyện và dự đoán thời điểm xảy ra hành động trong tương lai.

Vì thế, sau đây là những công việc mà Khoá luận đã hoàn thành:

- Chuẩn hoá bộ dữ liệu và xây dựng mô hình dự đoán chuỗi hành động của người dùng trong tương lai. Mô hình được đề tài sử dụng đã có tỷ lệ dự đoán chính xác cao hơn hầu hết các phương pháp baseline. Đặc biệt, khả năng phân lớp dữ liệu của mô hình đề tài có hiệu suất đáng kể và có thể giải quyết bài toán phân lớp đa biến.
- Rút ngắn thời gian huấn luyện và dự đoán. Từ đó, tiết kiệm chi phí thực nghiệm và cải tiến độ hiệu quả của mô hình.

Ngoài ra, đề tài của chúng tôi đã được chấp nhận tại Hội nghị Khoa học Trường Đại học Khoa học tự nhiên – Đại học quốc gia thành phố Hồ Chí Minh lần thứ 12, vào tháng 12 năm 2020.

6.2. Hướng phát triển

Mặc dù mô hình được sử dụng trong đề tài đã có bước cải tiến đáng kể về mặt thời gian dự đoán, tuy nhiên kết quả dự đoán vẫn chưa thực sự cao và chỉ cao hơn các phương pháp baseline từ 3% đến 14%. Do đó, trong tương lai, chúng tôi dự định sẽ nâng cao độ dự đoán chính xác của mô hình bằng việc thu thập thêm nhiều dữ liệu liên quan và tinh chỉnh tham số mô hình sao cho phù hợp với tính chất của bộ dữ liệu nhất.

Ngoài ra, việc dự đoán thành thời gian diễn ra hành động cũng có ý nghĩa rất to lớn trong cuộc sống hiện nay. Vậy nên, chúng tôi cũng sẽ hướng tới việc sử dụng các mô hình, thuật toán hồi quy để dự đoán thành công thời gian mà người dùng sẽ thực hiện bất kỳ một hành động nào đó tại một thời điểm cụ thể.

TÀI LIỆU THAM KHẢO

[1] Takeshi Kurashima, Tim Althoff and Jure Leskovec, "Modeling Interdependent and Periodic Real-World Action Sequences," *WWW,* 2018.

[2] P. Gorniak and D. Poole, "Predicting future user actions by observing unmodified applications," *AAAI,* 2000.

[3] E. Agichtein, E. Brill, and S. Dumais, "Improving web search ranking by incorporating user behavior information," *SIGIR,* 2006.

[4] F. Kooti, K. Lerman, L. M. Aiello, M. Grbovic, N. Djuric, and V. Radosavljevic, "Portrait of an online shopper: Understanding and predicting consumer behavior," *WSDM,* 2016.

[5] N. Du, H. Dai, R. Trivedi, U. Upadhyay, M. Gomez-Rodriguez, and L. Song, "Recurrent marked temporal point processes: Em- bedding event history to vector," *KDD,* 2016.

[6] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation,* 1997.

[7] W. Trouleau, A. Ashkan, W. Ding, and B. Eriksson, "Just one more: Modeling binge watching behavior," *KDD,* 2016.

[8] K. Kapoor, K. Subbian, J. Srivastava, and P. Schrater, "Just in time recommendations: Modeling the dynamics of boredom in activity streams," *WSDM,* 2015.

[9] J. Teevan, E. Adar, R. Jones, and M. Potts, "History repeats itself: Repeat queries in Yahoo's logs," *SIGIR,* 2006.

[10] D. R. Cox and V. Isham, Point processes, 1980.

[11] D. R. C. a. V. Isham, "Spectra of some self-exciting and mutually exciting point processes," *Biometrika,* 1971.

[12] T. Iwata, A. Shah, and Z. Ghahramani, "Discovering latent influence in online social activities via shared cascade poisson process," *KDD,* 2013.

[13] Y. Tanaka, T. Kurashima, Y. Fujiwara, T. Iwata, and H. Sawada, “Inferring latent triggers of purchases with consideration of social effects and media advertisements,” *WSDM,* 2016.

[14] K. Zhou, H. Zha, and L. Song, “Learning triggering kernels for multi-dimensional hawkes processes,” *ICML,* p. 1301– 1309, 2013.

[15] M. Farajtabar, Y. Wang, M. G. Rodriguez, S. Li, H. Zha, and L. Song, “Coevolve: A joint point process model for information diffusion and network co-evolution,” *NIPS,* 2015.

[16] N. Du, M. Farajtabar, A. Ahmed, A. J. Smola, and L. Song, “Dirichlet-hawkes processes with applications to clustering continuous-time document streams,” *KDD,* 2015.

[17] K. Kapoor, M. Sun, J. Srivastava, and T. Ye, “A hazard based approach to user return time prediction,” *KDD,* 2014.

[18] M. Haugh, “ The EM Algorithm,” *IEOR E4570: Machine Learning for OR&FE,* 2015.

[19] T. Q. Long, “Thuật toán Cực đại hóa Kì vọng (EM),” *Bài giảng Khoa Công nghệ Thông tin Trường Đại học Công nghệ,* 30 03 2016.

[20] Williams, Carl Eduard Rasmussen and Christopher K.I., Gaussian Processes for Machine Learning, MIT Press, 2006.

[21] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2013.

[22] Marian-Andrei Rizoiu, Young Lee, Swapnil Mishra, Lexing Xie, A Tutorial on Hawkes Processes for Events in Social Media.

[23] A. G. Hawkes, Spectra of some self-exciting and mutually exciting point processes, Biometrika, 1971.

[24] Ayşe Kızılersü, Markus Kreer and Anthony W. Thomas, The Weibull distribution, 2018.

[25] Rinne, H, The Weibull Distribution: A Handbook., Boca Raton, FL: CRC Press, 2009.

[26] A. Géron, Hands-On Machine Learning with Scikit-Learn and TensorFlow, O'Reilly Media, 2017.

[27] A. Ng, "Part II: Classification and logistic regression," trong *CS229 Lecture notes*.

[28] Hà Quang Thụy, Nguyễn Hà Nam, Nguyễn Trí Thành , "Giáo trình khai phá dữ liệu"".

[29] "Phần 1: Phân loại Naive Bayes (Lý thuyết)," Trí tuệ nhân tạo, 27 6 2019. [Trực tuyến]. Available: https://trituenhantao.io/kien-thuc/phan-1-phan-loai-naive-bayes-ly-thuyet/. [Đã truy cập 21 7 2021].

[30] N. T. A. Thư, "Slide bài giảng Phân lớp dữ liệu," 2020, p. 26.

[31] A. Navlani, "DataCamp," 2nd August 2018. [Trực tuyến]. Available: https://www.datacamp.com/community/tutorials/k-nearest-neighbor-classification-scikit-learn.

[32] Marina Sokolova and Guy Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management ,* tập 45, số 4, pp. 427-437, 2009.

[33] S. Narkhede, "Towards Data Science: Understanding AUC - ROC Curve," 26 6 2018. [Trực tuyến]. Available: https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5.

[34] M. L. c. bản, "Machine Learning cơ bản," Funda, 8 8 2017. [Trực tuyến]. Available: https://machinelearningcoban.com/2017/08/08/nbc/.