

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM  
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG  
KHOA CÔNG NGHỆ THÔNG TIN**



**VÕ ĐÌNH MINH TRÍ - 51900641  
NGÔ ĐỨC ANH TUẤN - 51800951**

**TÌM HIỂU BAYESIAN  
MINDSPONGE FRAMEWORK  
ANALYTICS VÀ ỨNG DỤNG**

**DỰ ÁN CÔNG NGHỆ THÔNG TIN 2**

**KHOA HỌC MÁY TÍNH**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2025**

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM  
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG  
KHOA CÔNG NGHỆ THÔNG TIN**



**VÕ ĐÌNH MINH TRÍ - 51900641  
NGÔ ĐỨC ANH TUẤN - 51800951**

# **TÌM HIỂU BAYESIAN MINDSPONGE ANALYTICS VÀ ỨNG DỤNG**

**DỰ ÁN CÔNG NGHỆ THÔNG TIN 2**

**KHOA HỌC MÁY TÍNH**

Người hướng dẫn  
**Th.S. Nguyễn Quốc Bình**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2025**

## LỜI CẢM ƠN

Chúng em xin chân thành cảm ơn Thầy Nguyễn Quốc Bình, giảng viên Khoa Công nghệ Thông tin, Trường Đại học Tôn Đức Thắng, người đã tận tình hướng dẫn và hỗ trợ em trong quá trình thực hiện bài báo cáo dự án Công nghệ Thông tin 2.

Nhờ sự chỉ dẫn tận tâm của Thầy, em đã có cơ hội tiếp cận, tìm hiểu và ứng dụng những kiến thức quan trọng vào thực tế, giúp em nâng cao kỹ năng phân tích, lập trình và phát triển tư duy khoa học. Những góp ý và định hướng quý báu từ Thầy không chỉ giúp em hoàn thành bài báo cáo một cách tốt nhất mà còn là hành trang quan trọng cho chặng đường học tập và làm việc sau này.

Một lần nữa, em xin gửi lời tri ân sâu sắc đến Thầy vì những kiến thức, sự động viên và sự hỗ trợ quý báu trong suốt quá trình học tập và nghiên cứu. Em hy vọng sẽ tiếp tục nhận được sự hướng dẫn của Thầy trong những chặng đường tiếp theo.

*TP. Hồ Chí Minh, ngày 10 tháng 02 năm 2025*

*Tác giả*

*(Ký tên và ghi rõ họ tên)*

Võ Đình Minh Trí

Ngô Đức Anh Tuấn

## CÔNG TRÌNH ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Chúng tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi và được sự hướng dẫn khoa học của **Th.S. Nguyễn Quốc Bình**. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong Dự án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung Dự án của mình.** Trường Đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày 10 tháng 02 năm 2025*

*Tác giả*

*(Ký tên và ghi rõ họ tên)*

*Võ Đình Minh Trí*

*Ngô Đức Anh Tuấn*

# TÌM HIỂU BAYESIAN MINDSPONGE ANALYTICS VÀ ỨNG DỤNG TÓM TẮT

Dự án này tập trung vào việc tìm hiểu và ứng dụng **Bayesian Mindsponge Framework (BMF)** trong phân tích dữ liệu, với mục tiêu áp dụng các phương pháp suy luận Bayesian để cải thiện quá trình phân tích và dự đoán trong các lĩnh vực như khoa học xã hội, kinh tế, hoặc phân tích dữ liệu lớn. Dự án bao gồm các bước từ việc lựa chọn công cụ lập trình phù hợp đến việc thu thập và xử lý dữ liệu, xây dựng mô hình Bayesian Mindsponge, và cuối cùng là đánh giá hiệu quả của mô hình.

Trong giai đoạn đầu của dự án, công cụ lập trình **Python** được lựa chọn nhờ vào khả năng hỗ trợ mạnh mẽ trong việc triển khai các mô hình Bayesian, cùng với các thư viện như **PyMC3** và **TensorFlow Probability** giúp thực hiện các phép tính phức tạp. Sau khi chọn công cụ lập trình, bước tiếp theo là thu thập dữ liệu phù hợp để áp dụng vào mô hình. Dữ liệu có thể được lấy từ các nguồn mở, khảo sát hoặc nghiên cứu trước đó, và phải trải qua quá trình tiền xử lý để làm sạch và chuẩn hóa trước khi đưa vào phân tích.

Sau khi dữ liệu đã được chuẩn bị, mô hình **Bayesian Mindsponge** sẽ được xây dựng, trong đó các tham số và biến quan sát được xác định rõ ràng. Phương pháp suy luận Bayesian sẽ được áp dụng để cập nhật xác suất và đưa ra dự đoán dựa trên dữ liệu quan sát. Các thuật toán như **Markov Chain Monte Carlo (MCMC)** sẽ được sử dụng để ước lượng các tham số và tối ưu hóa mô hình. Cuối cùng, mô hình sẽ được đánh giá thông qua các tiêu chí như độ chính xác, khả năng giải thích và tính hiệu quả trong việc phân tích và dự đoán.

Dự án không chỉ mang lại cái nhìn sâu sắc về **Bayesian Mindsponge Framework**, mà còn giúp người tham gia làm quen với việc ứng dụng các phương

pháp Bayesian vào thực tế, qua đó nâng cao khả năng phân tích và xử lý dữ liệu trong các lĩnh vực nghiên cứu khác nhau.

## MỤC LỤC

<b>DANH MỤC HÌNH VẼ .....</b>	<b>5</b>
<b>DANH MỤC BẢNG BIỂU .....</b>	<b>7</b>
<b>CHƯƠNG 1. MỞ ĐẦU VÀ TỔNG QUAN ĐỀ TÀI.....</b>	<b>8</b>
1.1 Lý do chọn đề tài.....	8
1.2 Mục tiêu thực hiện đề tài.....	8
<b>CHƯƠNG 2. LÝ THUYẾT VỀ BMF.....</b>	<b>10</b>
2.1 Mô hình Bayesian và nguyên tắc hoạt động .....	10
2.2 Mindsponge Framework .....	12
2.3 BMF .....	13
<b>CHƯƠNG 3. CƠ SỞ LÝ THUYẾT.....</b>	<b>15</b>
3.1 Python .....	15
3.2 TF-IDF .....	19
3.2.1 <i>TF - Tern Frequency</i> .....	19
3.2.2 <i>IDF - Inverse Document Frequency</i> .....	20
3.2.3 <i>Kết luận</i> .....	21
<b>CHƯƠNG 4. CÔNG CỤ VÀ PHƯƠNG PHÁP .....</b>	<b>22</b>
4.1 Công cụ .....	22
4.2 Quy trình xây dựng mô hình BMF.....	22
4.2.1 <i>Chọn dữ liệu nghiên cứu</i> .....	22
4.2.2 <i>Định nghĩa biến quan sát và tham số</i> .....	23
4.2.3 <i>Phân tích dữ liệu bằng phương pháp Bayesian</i> .....	23
4.2.4 <i>Đánh giá hiệu quả và độ chính xác của mô hình</i> .....	23

<b>CHƯƠNG 5. ỨNG DỤNG THỰC TIỄN.....</b>	<b>25</b>
5.1 Chọn data .....	25
5.1.1 Phân tích data .....	25
5.1.2 Tiền xử lý dữ liệu .....	31
5.1.3 Định nghĩa các biến và tham số của dữ liệu.....	31
5.2 Áp dụng công thức Bayesian để phân tích dữ liệu.....	35
5.2.1 Naive Bayes.....	35
5.2.2 Random Forest (Non-Bayes).....	37
5.3 So sánh và đánh giá.....	40
5.4 Naive Bayes và Random Forest chưa phải là BMF? .....	41
5.4.1 Naive Bayes.....	41
5.4.2 Random Forest.....	42
5.4.3 Tiếp cận BMF .....	43
5.5 Kết luận .....	47
5.5.1 Ưu điểm.....	47
5.5.2 Nhược điểm.....	48
<b>CHƯƠNG 6. ĐÁNH GIÁ VÀ KẾT LUẬN .....</b>	<b>49</b>
6.1 Điểm mạnh của BMF .....	49
6.2 Điểm yếu của BMF .....	49
6.3 Khắc phục.....	50
6.4 Tổng kết .....	50
<b>TÀI LIỆU THAM KHẢO .....</b>	<b>53</b>



## DANH MỤC HÌNH VẼ

Hình 2.1: Ví dụ về tư duy Bayesian.....	11
Hình 2.2: Cơ chế Mindsponge .....	12
Hình 3.1: Khảo sát Kdnuggets Analytics, Khoa học dữ liệu, Phần mềm học máy năm 2016 – 2018 .....	15
Hình 3.2: Ứng dụng của Python.....	16
Hình 3.3: Top các ngôn ngữ lập trình phổ biến nhất năm 2024 .....	18
Hình 5.1: Biểu đồ số lượng video theo thể loại .....	27
Hình 5.2: <i>Code vẽ biểu đồ phân bố lượng likes, dislikes, comment_count</i> .....	28
Hình 5.3: biểu đồ so sánh phân bố likes, dislike, comment_count .....	29
Hình 5.4: Heatmap tương quan giữa các biến.....	30
Hình 5.5: Code điền dữ liệu cho các biến bị thiếu .....	31
Hình 5.6: Biến hành vi .....	32
Hình 5.7: Biến cảm xúc.....	32
Hình 5.8: Biến tri thức áp dụng TF-IDF .....	33
Hình 5.9: Kết hợp các biến tri thức sau khi dùng TF-IDF .....	34
Hình 5.10: Code kết hợp các biến tri thức, cảm xúc, hành vi lại với nhau.....	34
Hình 5.11: Xác định biến mục tiêu .....	35
Hình 5.12: Dùng mô hình Naive Bayes để huấn luyện.....	35
Hình 5.13: Kết quả đánh giá mô hình Naïve Bayes.....	36
Hình 5.14: Huấn luyện Random Forest .....	38
Hình 5.15: Đánh giá mô hình.....	39
Hình 5.16: Lọc các biến .....	44

Hình 5.17: Chuyển đổi dữ liệu thành ‘category’ .....	45
Hình 5.18: Cấu trúc mô hình phù hợp.....	45
Hình 5.19: Sử dụng Bayesian Estimator trong pgmpy .....	46
Hình 5.20: Kiểm tra tổng xác suất .....	46
Hình 5.21: Kết quả kiểm tra.....	47

## **DANH MỤC BẢNG BIỂU**

Table 5.1 So sánh kết quả đánh giá mô hình .....	40
Bảng 5.2: Biến và các mối quan hệ Bayesian .....	43

# CHƯƠNG 1. MỞ ĐẦU VÀ TỔNG QUAN ĐỀ TÀI

## 1.1 Lý do chọn đề tài

Hiện nay, khoa học dữ liệu đang phát triển một cách mạnh mẽ, việc ứng dụng các phương pháp khoa học, học máy để tìm hiểu về hành vi, cảm xúc của con người trong các lĩnh vực kinh tế, tâm lý, ... là điều vô cùng cần thiết.

Khi thấy được đề tài “Tìm hiểu Bayesian Mindsponge Framework analytics và ứng dụng” của thầy Nguyễn Quốc Bình, chúng em lập tức đăng ký ngay, vì chúng em biết được rằng Bayesian Mindsponge Framework (BMF) là một phương pháp tiếp cận kết hợp giữa Mindsponge và Bayesian nhằm phân tích và diễn giải dữ liệu trong lĩnh vực khoa học xã hội và hành vi con người.

Việc nghiên cứu BMF không chỉ cung cấp thêm được kiến thức về toán học và xác suất thống kê, mà tăng khả năng ứng dụng học máy trong việc phân tích dữ liệu ở các lĩnh vực thực tế. BMF còn cung cấp khả năng phân tích các dữ liệu không chắc chắn nhằm đưa ra các dự đoán hợp lý dựa trên thông tin sẵn có. Điều này khá hữu ích trong việc nghiên cứu về tâm lý học, xã hội học, và các lĩnh vực khác liên quan tới con người.

Ngoài ra, nghiên cứu BMF còn giúp tiếp cận các công cụ lập trình mạnh mẽ hiện nay, điển hình là Python. Việc áp dụng các phương pháp hiện đại như BMF không chỉ góp phần nâng cao kỹ năng lập trình và phân tích dữ liệu của cá nhân mà còn có tiềm năng đóng góp vào sự phát triển của các nghiên cứu khoa học và ứng dụng thực tế.

## 1.2 Mục tiêu thực hiện đề tài

Mục tiêu chính của đề tài là nghiên cứu cơ bản về Bayesian Mindsponge Framework (BMF) và nguyên lý hoạt động của nó, bao gồm cách kết hợp giữa tư duy Bayesian và mô hình Mindsponge, từ đó phân tích các đặc điểm nổi bật cũng như sự khác biệt của BMF so với các phương pháp phân tích dữ liệu truyền thống.

Bên cạnh đó, đề tài hướng đến việc khám phá khả năng ứng dụng của BMF trong nhiều lĩnh vực như khoa học xã hội, tâm lý học, kinh tế học, giáo dục hoặc phân tích dữ liệu lớn, nhằm đánh giá tính hiệu quả của phương pháp này trong việc phân tích hành vi, cảm xúc của con người, dự đoán xu hướng và xử lý dữ liệu không chắc chắn.

Ngoài việc nghiên cứu BMF, đề tài còn hướng đến việc phát triển khả năng lập trình, khai thác các mô hình Bayesian trong thực tế bằng các ngôn ngữ lập trình mạnh mẽ hiện nay như Python, R, ... Đề tài giúp phát triển kỹ năng phân tích dữ liệu, tiền xử lý dữ liệu, xây dựng các mô hình học máy, huấn luyện và đánh giá kết quả.

Đặc biệt, đề tài sẽ đưa ra các ví dụ minh họa cụ thể về cách sử dụng BMF analytics, bao gồm quy trình phân tích dữ liệu, diễn giải kết quả và đánh giá ý nghĩa thực tiễn của phương pháp này. Thông qua quá trình nghiên cứu và thực hành, người thực hiện không chỉ nắm vững kiến thức về BMF mà còn có thể ứng dụng nó vào thực tế, góp phần nâng cao hiệu quả phân tích dữ liệu và hỗ trợ ra quyết định một cách khoa học hơn.

## CHƯƠNG 2. LÝ THUYẾT VỀ BMF

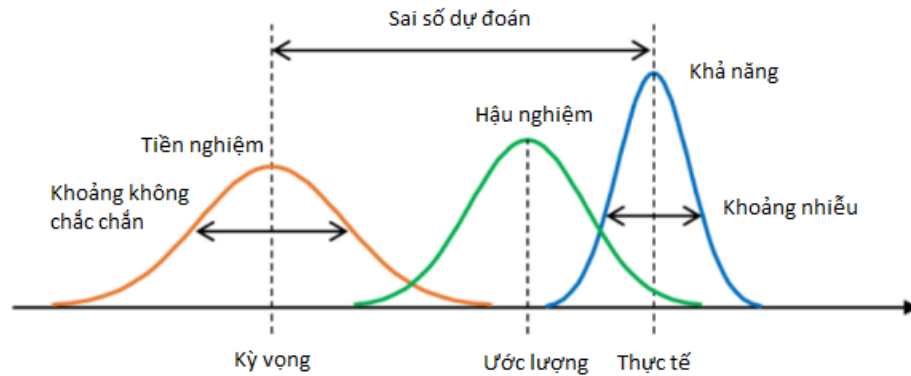
### 2.1 Mô hình Bayesian và nguyên tắc hoạt động

Tư duy Bayesian đưa ra xác suất hậu nghiệm (Posterior Probability) là hệ quả của hai tiền đề:

- Xác suất tiên nghiệm (Prior Probability).
- Hàm khả năng (Likelihood Function).

$$P(\theta|\mathbf{D}) = \frac{P(\mathbf{D}|\theta)P(\theta)}{P(\mathbf{D})}$$

- $\theta$  là tham số thể hiện giả thuyết cần đánh giá, hay là hệ số cần xác định mà xác suất của nó chịu ảnh hưởng từ dữ liệu quan sát.
- $\mathbf{D}$  là dữ liệu quan sát được, số liệu thu được từ thực tế.
- $P(\theta)$  là xác suất tiên nghiệm, niềm tin hay các bằng chứng trước đây liên quan tới giả thuyết (Prior Probability).
- $P(\mathbf{D}|\theta)$  là xác suất quan sát được dữ liệu  $\mathbf{D}$  với điều kiện  $\theta$  (likelihood).
- $P(\mathbf{D})$  được gọi là khả năng biên (Marginal Likelihood), một hằng số không phụ thuộc vào tham số  $\theta$ .
- $P(\theta|\mathbf{D})$  là xác suất hậu nghiệm, luôn tỷ lệ với xác suất tiên nghiệm và xác suất hậu nghiệm (Posterior Probability).



Hình 2.1: Ví dụ về tư duy Bayesian

(Nguồn: Hình trực quan được lấy về từ Yanagisawa, Kawamata và Ueda dưới giấy phép Creative Commons Attributions (CC-BY))

Tư duy Bayes sử dụng các khoảng đáng tin cậy xem các tham số ước tính là các biến ngẫu nhiên và giới hạn của chúng là cố định. Nếu so sánh với phương pháp tần số sử dụng khoảng tin cậy coi tham số ước tính là hằng số và giới hạn của chúng là các biến ngẫu nhiên, thì suy luận Bayes có nhiều lợi thế lý thuyết lớn hơn.

Tư duy Bayes cung cấp ước lượng chính xác hơn với tập mẫu có kích thước nhỏ so với phương pháp tần số, giúp giảm thiểu chi phí đầu tư vào các bộ dữ liệu lớn.

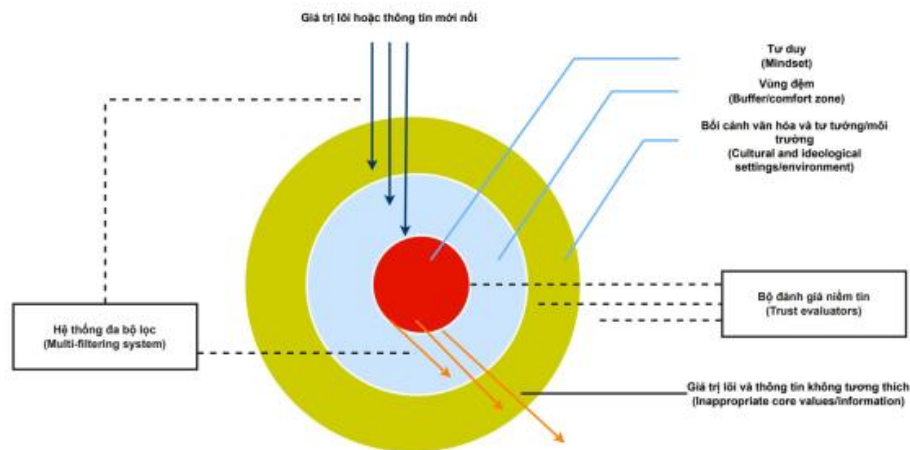
Tư duy Bayes không phân biệt giữa dữ liệu chưa được quan sát và các tham số chưa biết, và sử dụng các giá trị đã biết và chưa biết dựa trên xác suất. Giả sử kích thước mẫu có thể tăng thêm với các giá trị ước lượng và phương pháp kiểm định dưới giới hạn từ  $n$  đến  $\infty$ , kết quả ước lượng của suy luận Bayes chỉ dựa trên dữ liệu đã có.

## 2.2 Mindsponge Framework

Là một mô hình nhận thức tập trung vào cách con người thu nhận, chọn lọc và lưu trữ thông tin. Mindsponge sử dụng cơ chế màng lọc, để giải thích các hành vi và tâm lý của các cá nhân hay tập thể.

Thuật ngữ mindsponge ẩn dụ cho việc tâm trí và tư duy (mind) của con người như một chiếc bọt biển (sponge) có khả năng hấp thu các giá trị mới và loại bỏ các giá trị không tương thích với các giá trị cốt lõi của nó.

Cơ chế mindsponge là một quá trình động và phức tạp, được biểu diễn qua một sơ đồ khái niệm dạng vòng tròn biểu thị tâm trí và môi trường xung quanh.



Hình 2.2: Cơ chế Mindsponge

Nguồn: (Wikipedia – Cơ chế Mindsponge)

Sơ đồ bao gồm năm thành phần chính: Tư duy, Vùng đệm, Hệ thống đa bộ lọc, Bối cảnh văn hóa và tư tưởng, Giá trị văn hóa/ thông tin .

Phần ngoài cùng – màu vàng – thể hiện bối cảnh văn hóa và tư tưởng mà cá nhân đang tồn tại. Phần màu xanh tiếp theo là vùng đệm, bao gồm các giá trị trong tâm trí nhưng không phải giá trị cốt lõi. Phần này có hai chức năng cơ bản: bảo vệ vùng tư duy khỏi những cú sốc từ bên ngoài khi môi trường thay đổi, và là nơi hệ thống lọc đa cấp bắt đầu hoạt động để đánh giá tính thích hợp của các giá trị mới được hấp thu. Phần trong cùng có màu đỏ thể hiện cho tư duy hay tập hợp các giá



trị cốt lõi. Các mảng trắng giữa các phần màu biểu thị điểm đánh giá hoặc điểm lọc để dễ hình dung hơn.

## 2.3 BMF

Là một công cụ được giới thiệu trong lĩnh vực phân tích dữ liệu và nghiên cứu khoa học xã hội để mô hình hóa và giải thích các hành vi nhận thức, ra quyết định, trao đổi thông tin trong một môi trường phức tạp.

Xương sống của BMF:

- Hệ Lý thuyết quản trị tri thức Serendipity Mindsponge 3D
- Bộ ba triết lý khoa học về minh bạch, chi phí, sự chủ động
- Khoa học mở 3O (open data, open review, open dialogue)
- Phần mềm bayesvl
- Văn hóa Mindsponge

BMF được kết hợp từ cơ chế mindsponge – có khả năng thể hiện độ phức tạp và động lực của suy nghĩ con người, được dùng để dựng nên các mô hình lý thuyết, với suy luận Bayes có độ linh hoạt cao, cho phép các nhà nghiên cứu xác định độ phù hợp của các mô hình này. Nhờ sự tương thích cao giữa cơ chế mindsponge và suy luận Bayes, ta thấy được 5 điểm mạnh đặc trưng của BMF. Nhờ vào đó, BMF analytics đã được áp dụng để khám phá ra các vấn đề xã hội, tâm lý và hành vi đa dạng trong nhiều lĩnh vực như sức khỏe tinh thần, giáo dục, tâm lý học, y tế... 5 điểm mạnh đó là:

1) Chủ quan có kiểm soát: Với việc cả suy luận Bayes và cơ chế mindsponge đều mang tính chủ quan. Với mindsponge, nó đến từ kết quả của quá trình tư duy tùy theo ngữ cảnh. Còn suy luận Bayes thì được xây dựng dựa trên xác suất chủ quan. BMF giúp các nhà khoa học cân nhắc các yếu tố chủ quan đó và vẫn giữ được tính chính xác trong việc xây dựng mô hình và phân tích thống kê.

2) Linh hoạt và tinh gọn: BMF có tính linh hoạt cao trong việc nghiên cứu các quá trình phức tạp của tư duy con người. Cơ chế mindsponge giúp xây dựng

các mô hình phản ánh quá trình xử lý thông tin, trong khi suy luận bayes, kết hợp với thuật toán MCMC, cho phép điều chỉnh các mô hình phức tạp. BMF ưu tiên tính parsimony (đơn giản mà hiệu quả) để tăng khả năng giải thích và dự đoán. Cơ chế mindsponge hỗ trợ xác định ranh giới nghiên cứu, giúp tối ưu hóa quá trình xây dựng mô hình.

3) Phân tích phân cấp linh hoạt: BMF có khả năng xử lý hiệu quả các vấn đề phân cấp bằng cách sử dụng cơ chế mindsponge để xây dựng mô hình phản ánh sự khác biệt ở các cấp độ khác nhau (cá nhân, tập thể). Phương pháp này kết hợp suy luận Bayes và thuật toán MCMC để gán phân phối xác suất cho các tham số hồi quy, giúp kiểm tra và điều chỉnh các mô hình phân cấp phức tạp, bao gồm cả phi tuyến tính.

4) Tối ưu hóa sử dụng thông tin tiên nghiệm: BMF tận dụng cơ chế mindsponge để xác định và biện luận cho các xác suất tiên nghiệm (prior) hợp lý trong suy luận Bayes, giúp tăng độ chính xác của mô hình, đặc biệt với tập dữ liệu nhỏ. Điều này không chỉ giảm chi phí nghiên cứu mà còn hỗ trợ các nghiên cứu khoa học tại những nơi có nguồn lực hạn chế. Ngoài ra, việc tích hợp prior còn giúp giải quyết vấn đề đa cộng tuyến hiệu quả hơn so với phương pháp hồi quy Ridge.

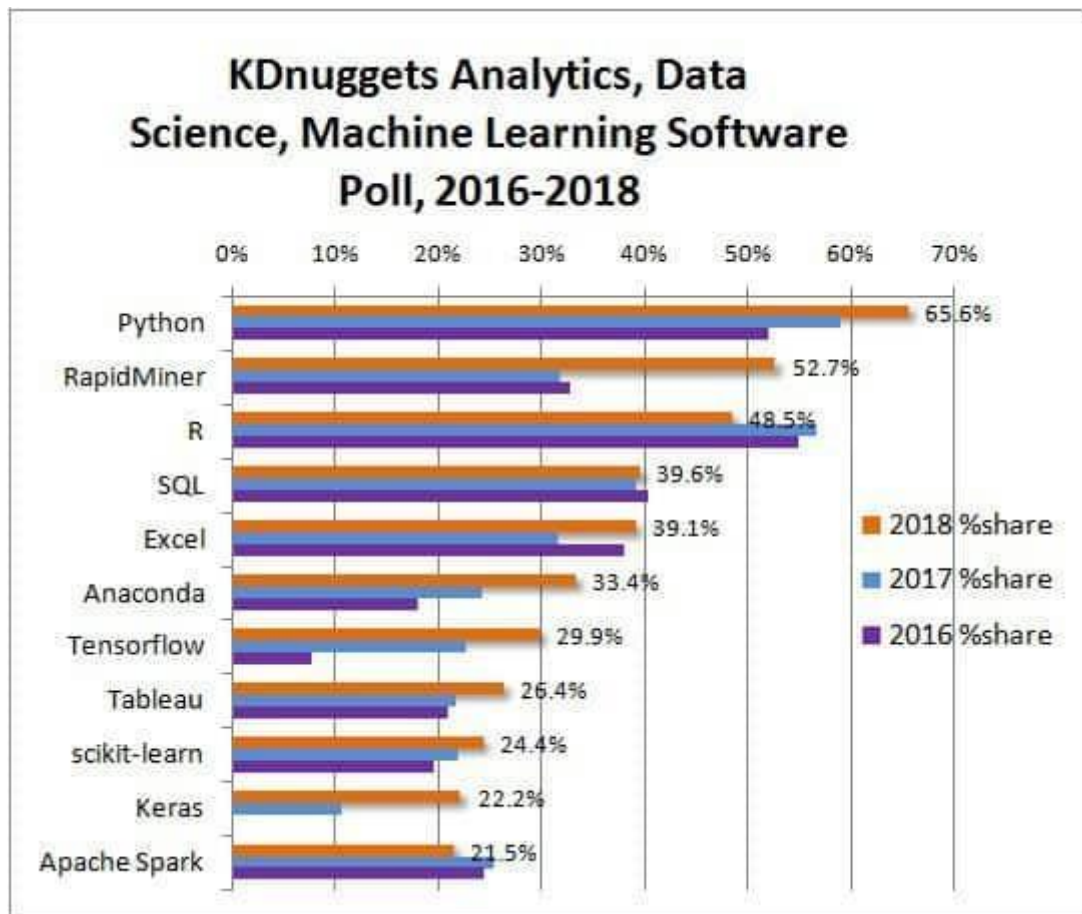
5) Khả năng cập nhật linh hoạt: Cả cơ chế mindsponge và suy luận Bayes đều có khả năng cập nhật liên tục, giúp các nhà nghiên cứu phân tích các hiện tượng tâm lý theo thời gian trong bối cảnh thông tin luôn thay đổi. Điều này cho phép đánh giá tác động của yếu tố hoàn cảnh đối với các quá trình tư duy, góp phần giải quyết cuộc khủng hoảng tái sản xuất (reproducibility crisis) trong tâm lý học.

## CHƯƠNG 3. CƠ SỞ LÝ THUYẾT

### 3.1 Python

Vào những năm cuối thập niên 80, đầu những năm thập niên 90, Guido van Rossum đã phát minh ra ngôn ngữ Python. Python là ngôn ngữ bậc cao với muôn vàn tính năng hỗ trợ cho rất nhiều ứng dụng khác nhau.

Python đang dần có vai trò quan trọng trong lập trình nói chung và trong ngành khoa học máy tính, khoa học dữ liệu nói riêng. Python cung cấp nhiều thư viện phục vụ cho việc xử lý dữ liệu như NumPy dùng để tính toán, Pandas dùng để phân tích dữ liệu, Matplotlib dùng để trực quan hóa dữ liệu.



Hình 3.1: Khảo sát Kdnuggets Analytics, Khoa học dữ liệu, Phần mềm học máy năm 2016 – 2018

Nguồn: (Codelearn – Python cơ bản)

Ngoài nghiên cứu data, học máy, Python còn được dùng để phát triển web. Python cung cấp cho lập trình viên hai frameworks là Django và Flask. Các đơn vị đình đám dùng Python để phát triển web của họ có thể kể đến như Dropbox, Netflix, Spotify, Instagram, ...

Không chỉ Web, Python còn được dùng để phát triển GUI – giao diện đồ họa bằng thư viện Tkinter, phát triển game với thư viện Pygame, ...

Ngoài ra, Python còn được dùng để phát triển trí tuệ nhân tạo, phân tích dữ liệu, ERP, ... và vô vàn lĩnh vực khác.



Hình 3.2: Ứng dụng của Python

Nguồn: (Ứng dụng của Python phổ biến nhất trong thực tế)

Các câu lệnh trong python:

- =: phép gán
- If – else: câu lệnh điều kiện
- For: câu lệnh vòng lặp
- Def: hàm
- Break: dùng để thoát khỏi vòng lặp hoặc câu điều kiện khi đã thỏa điều kiện.

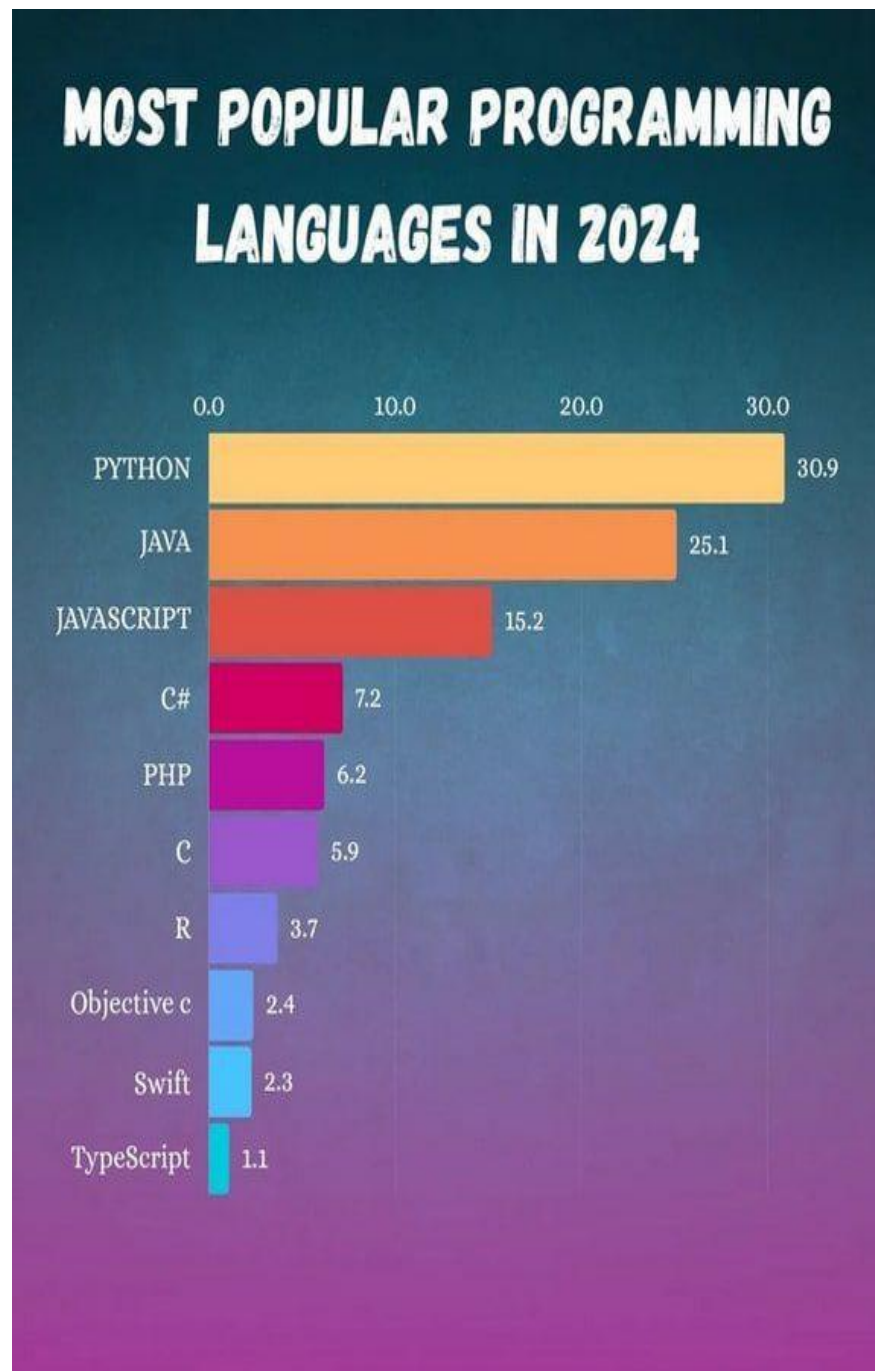
- Return: trả về kết quả mong muốn
- ...

Trên là một số ít trong nhiều câu lệnh thường dùng trong Python mà lập trình viên sẽ thường xuyên tương tác.

Ngoài ra còn có các kiểu dữ liệu trong python như:

- Int: kiểu số nguyên.
- Float: kiểu số thập phân
- Bool: kiểu logic.
- List: kiểu danh sách.
- ...

Từ năm 2003, Python đã trở nên phổ biến trong giới lập trình, tới hiện nay, Python đã trở nên vô cùng phổ biến và được dùng trong các tập đoàn, tổ chức lớn như NASA, Google, Yahoo, Facebook, ...



Hình 3.3: Top các ngôn ngữ lập trình phổ biến nhất năm 2024

Nguồn: (Python là gì? Tổng hợp kiến thức cho người mới bắt đầu.)

## 3.2 TF-IDF

Trong ngôn ngữ tự nhiên, luôn tồn tại một số các từ có tần suất xuất hiện rất cao, và đôi khi các từ đó lại không có tính quan trọng cao trong tập hợp các từ xuất hiện. Để làm cho tần suất của các từ trở nên cân bằng với nhau, TF-IDF là một phương pháp hợp lý để làm được điều này.

TF-IDF là phương pháp khá phổ biến, dùng để đánh giá độ quan trọng của từ trong một câu hay một văn bản, từ đó giúp chúng ta phân loại, tìm kiếm thông tin, hoặc chọn ra các từ khóa để có thể sử dụng cho các công việc như thị hiếu người dùng, cảm nhận người dùng về một sự vật, sự việc nào đó.

TF-IDF không chỉ tập trung vào tần suất xuất hiện của một từ trong văn bản, mà còn tập trung vào độ phổ biến của từ ngữ đó, từ đó mà có thể giảm đi sự ảnh hưởng của những từ xuất hiện rất nhiều nhưng không mang lại quá nhiều giá trị trong việc phân tích dữ liệu.

### 3.2.1 TF - Tern Frequency

TF (Tern Frequency) hay còn được gọi là “**tần suất xuất hiện của từ**”, được dùng để tính toán tần suất xuất hiện của một từ nào đó trong một văn bản cụ thể. TF giúp cho việc so sánh tần suất của một từ cụ thể trong các văn bản khác nhau trở nên cân bằng, điều này là rất cần thiết vì mỗi văn bản khác nhau sẽ có độ dài khác nhau, làm cho số lần xuất hiện của một từ cụ thể trong từng văn bản là không giống nhau.

Công thức cho TF là:

$$TF = \frac{t}{x}$$

Trong đó:

- t – Số lần xuất hiện của một từ trong văn bản
- x – Tổng số từ trong văn bản đó

Ví dụ ta có một câu sau:

‘Tôi rất thích ăn mì trộn và tôi ghét các món rau.’

Trong câu trên, để tính tần suất xuất hiện của từ “Tôi”, ta thấy từ “Tôi” xuất hiện 2 lần, và tổng số từ trong câu trên là 12, vậy áp dụng công thức tính ta có thể tính ra được tần suất xuất hiện của từ “Tôi” bằng  $\frac{2}{12} = 1/6 \approx 0.17$ .

Mặc dù có thể tính toán được tần suất xuất hiện của một từ trong văn bản, nhưng TF lại không thể đưa ra được mức độ phổ biến của từ đó. Vì vậy IF cần được kết hợp với IDF để làm được điều này.

### 3.2.2 IDF - Inverse Document Frequency

Để khắc phục nhược điểm mà TF mang lại, chúng ta cần kết hợp TF với IDF để có thể đo lường được sự phổ biến của từ trong toàn bộ tài liệu được cung cấp.

IDF (Inverse Document Frequency) được dùng để đánh giá mức quan trọng của một từ nào đó trong một văn bản cụ thể.

IDF cần thiết là vì có rất nhiều từ, mặc dù tần suất xuất hiện rất cao, nhưng lại không thật sự quan trọng vì các từ đó không có tính đặc trưng. Giả sử một từ có tần suất xuất hiện cao trong một văn bản nói riêng và nhiều văn bản nói chung, điều này nói lên rằng từ đó có độ phổ biến cao, thành ra tính đặc trưng của từ đó khá thấp.

Ta có thể dễ thấy nhất ở các trường hợp như:

- Các từ nối như ‘và’, ‘hoặc’, ‘nhưng’, ‘bởi vì’, ‘tuy nhiên’, ‘vì vậy’, ‘nếu’, ...
- Các giới từ như ‘Theo’, ‘đến’, ‘kể từ’, ‘trong khi’, ‘ở trong’, ‘ở ngoài’, ...
- Các từ chỉ định như ‘các’, ‘mấy’, ...

Các từ này tuy có tần suất xuất hiện rất cao, tuy nhiên, các từ này có trong câu hoặc không có trong câu cũng không quá quan trọng, và IDF sẽ làm việc để chỉ ra được mức độ quan trọng của các từ đó.

Công thức IDF được tính như sau:



$$\text{IDF} = \log \left( 1 + \frac{t}{x} \right)$$

Trong đó:

- $t$  – Tổng tài liệu, văn bản
- $x$  – tổng tài liệu, văn bản chứa từ cần kiểm tra

Ví dụ: ta có từ ‘ngựa’ xuất hiện trong khoản 10 trên 100 câu ngẫu nhiên, từ ‘và’ xuất hiện trong khoản 70 trên 100 câu ngẫu nhiên nào đó, áp dụng công thức ta có thể thấy mức quan trọng của từ ‘ngựa’ cao hơn so với từ ‘và’.

Qua ví dụ trên, ta có thể thấy được rằng, IDF giúp cho ta có thể xác định mức độ quan trọng của các từ đang kiểm tra để có thể làm giảm trọng số các từ không có tính quan trọng cao.

TF-IDF có tính ứng dụng cao trong việc tìm kiếm. TF-IDF được sử dụng để chỉ ra được các từ khóa mà người ta quan tâm, thường dùng nhất, dựa vào đó để đưa ra kết luận và đưa ra được kết quả mong muốn.

### 3.2.3 Kết luận

TF-IDF là một phương pháp hiệu quả để xử lý và phân tích các văn bản trong nhiều ứng dụng khác nhau. Nó giúp cân bằng giữa tần suất xuất hiện của các từ và mức độ phổ biến của chúng trong toàn bộ tài liệu được cung cấp, từ đó tìm ra những từ quan trọng và đặc trưng nhất. Bằng cách này, TF-IDF giúp cải thiện độ chính xác trong việc tìm kiếm thông tin, phân loại văn bản, và nhiều ứng dụng khác trong lĩnh vực xử lý ngôn ngữ tự nhiên.

Trong tương lai, các kỹ thuật cải tiến hoặc kết hợp TF-IDF với các mô hình học sâu có thể tạo ra các phương pháp phân tích văn bản mạnh mẽ và chính xác hơn, từ đó hỗ trợ tốt hơn trong việc xử lý và hiểu ngôn ngữ tự nhiên.

## CHƯƠNG 4. CÔNG CỤ VÀ PHƯƠNG PHÁP

### 4.1 Công cụ

Trong quá trình nghiên cứu và triển khai mô hình Bayesian Mindsponge Framework (BMF), việc lựa chọn công cụ phân tích phù hợp đóng vai trò quan trọng trong việc đảm bảo tính chính xác và hiệu quả của mô hình, cũng như sự quen thuộc của lập trình viên đối với công cụ, ngôn ngữ lập trình đó.

Python là một sự lựa chọn làm công cụ hợp lý để thực hiện phân tích dữ liệu và xây dựng mô hình Bayesian bởi tính linh hoạt và sự mạnh mẽ của nó trong việc xử lý dữ liệu thống kê.

Python sở hữu một hệ sinh thái phong phú gồm nhiều thư viện hỗ trợ Bayesian modeling như PyMC, TensorFlow Probability và scikit-learn, giúp tối ưu hóa quá trình triển khai mô hình.

Bên cạnh đó, Python cũng hỗ trợ trực quan hóa dữ liệu thông qua các thư viện như Matplotlib, Seaborn, giúp đánh giá kết quả một cách trực quan và dễ hiểu.

Việc sử dụng Python không chỉ giúp tự động hóa các bước tính toán mà còn tạo điều kiện thuận lợi cho việc thử nghiệm, kiểm tra và tinh chỉnh mô hình BMF một cách linh hoạt và hiệu quả.

### 4.2 Quy trình xây dựng mô hình BMF

#### 4.2.1 Chọn dữ liệu nghiên cứu

Dữ liệu đóng vai trò cốt lõi trong việc xây dựng mô hình. Tùy vào lĩnh vực nghiên cứu như khoa học xã hội, tâm lý học, kinh tế hoặc giáo dục, dữ liệu có thể được thu thập từ khảo sát, báo cáo thống kê hoặc các nguồn dữ liệu lớn. Dữ liệu này cần được tiền xử lý để loại bỏ các giá trị không hợp lệ, xử lý dữ liệu bị thiếu và chuẩn hóa nhằm đảm bảo độ tin cậy trước khi đưa vào mô hình.

#### ***4.2.2 Định nghĩa biến quan sát và tham số***

Sau khi thu thập dữ liệu, cần xác định các yếu tố quan trọng của mô hình.

Biến quan sát là những thông tin có thể đo lường trực tiếp từ dữ liệu, ví dụ như mức độ hài lòng của khách hàng, thói quen tiêu dùng hoặc điểm số của học sinh.

Tham số chưa biết là những yếu tố cần ước lượng, chẳng hạn như mức độ ảnh hưởng của một yếu tố đến quyết định của con người.

Phân phối tiên nghiệm thể hiện thông tin có sẵn trước khi quan sát dữ liệu, dựa trên kinh nghiệm hoặc các nghiên cứu trước đó.

Phân phối hậu nghiệm là kết quả cuối cùng sau khi dữ liệu được phân tích, phản ánh thông tin mới nhất về các tham số quan trọng trong mô hình.

#### ***4.2.3 Phân tích dữ liệu bằng phương pháp Bayesian***

Dữ liệu sau khi được chuẩn bị sẽ được đưa vào mô hình để phân tích theo phương pháp Bayesian. Quá trình này sử dụng thông tin sẵn có từ phân phối tiên nghiệm kết hợp với dữ liệu thực tế để điều chỉnh và cập nhật các tham số của mô hình. Các thuật toán phổ biến như Markov Chain Monte Carlo (MCMC) hoặc Variational Inference giúp thực hiện việc ước lượng và tối ưu hóa mô hình.

#### ***4.2.4 Đánh giá hiệu quả và độ chính xác của mô hình***

Sau khi mô hình được xây dựng, cần đánh giá mức độ phù hợp với dữ liệu thực tế. Một số phương pháp kiểm tra bao gồm:

Kiểm định hậu nghiệm, so sánh dự đoán của mô hình với dữ liệu thực tế để xem xét mức độ chính xác.

Tiêu chí đánh giá mô hình, như WAIC (Widely Applicable Information Criterion) và LOO-CV (Leave-One-Out Cross-Validation), giúp so sánh mô hình với các phương pháp khác.

Kiểm tra độ hội tụ, đảm bảo rằng thuật toán tối ưu hóa đã tìm ra kết quả ổn định và đáng tin cậy.

Bằng cách thực hiện đầy đủ các bước trên, mô hình Bayesian Mindsponge Framework không chỉ giúp phân tích dữ liệu một cách hiệu quả mà còn cung cấp những dự đoán và thông tin có giá trị trong nghiên cứu khoa học và ứng dụng thực tiễn.

## CHƯƠNG 5. ỨNG DỤNG THỰC TIỄN

### 5.1 Chọn data

#### 5.1.1 Phân tích data

Đầu tiên cần xác định được bộ data phù hợp cho việc xây dựng mô hình. Ở đây chúng ta sẽ chọn data có yếu tố phản ánh về xu hướng, hành vi của con người, hay là data đánh giá, phản hồi về một sự vật, sự việc, sự kiện nào đó.

Sau khi xem xét các nguồn dữ liệu uy tín, bộ dữ liệu “**Trending YouTube Video Statistics**” từ Kaggle đã được chọn. Đây là tập hợp dữ liệu bao gồm:

- Nhiều khu vực: Dữ liệu từ các quốc gia khác nhau, mỗi quốc gia được lưu trong một file CSV riêng biệt.
- Thời gian: Ghi lại thông tin về các video trending trong một khoảng thời gian dài.

Trong bộ data có các thành phần như mô tả, tiêu đề, tag, lượt thích, lượt không thích, số bình luận, id video, ...

Vì là nhiều file csv khác nhau, nên để thuận tiện trong việc phân tích, tiến hành gom chúng lại thành một data duy nhất để có thể dễ dàng phân tích hơn. Trong python, thư viện pandas có thể thực hiện điều này một cách dễ dàng.

Các thành phần thông tin chính:

- video\_id: ID duy nhất của video.
- trending\_date: Ngày video trở thành trending.
- title: Tiêu đề của video.
- channel\_title: Tên kênh phát hành video.
- category\_id: Mã thể loại của video.
- publish\_time: Thời gian phát hành video.
- tags: Thẻ liên quan đến video.
- views: Số lượt xem.
- likes: Số lượt thích.

- dislikes: Số lượt không thích.
- comment\_count: Số lượng bình luận.
- description: Mô tả chi tiết của video.

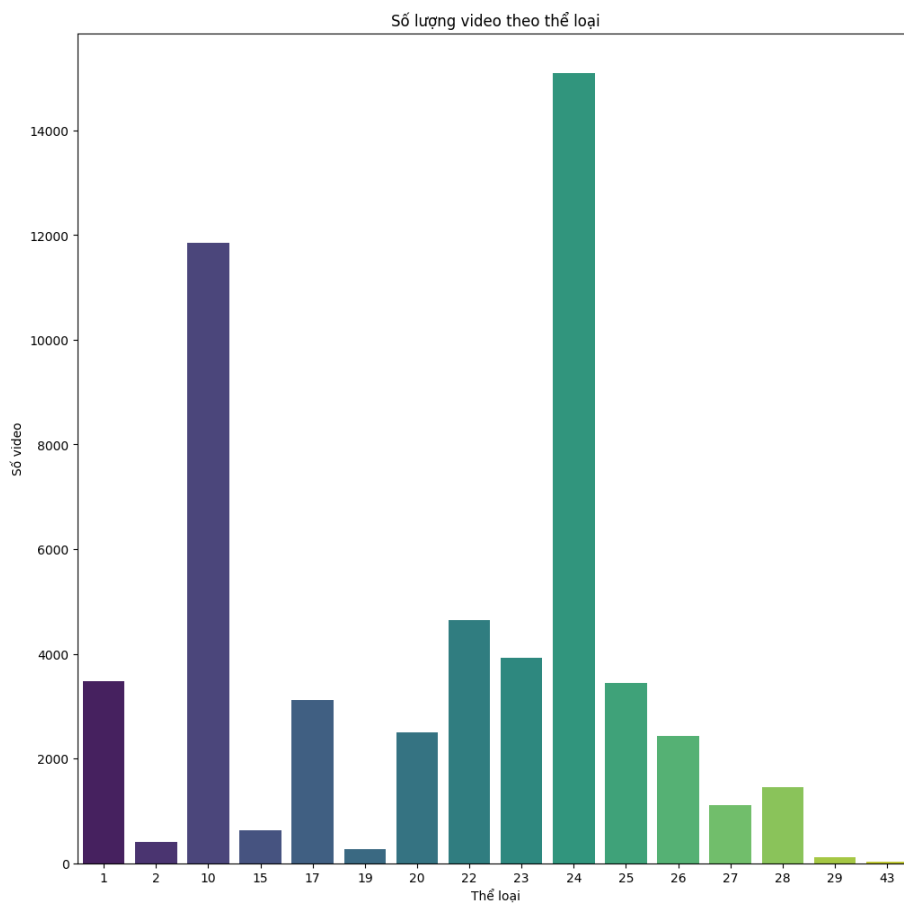
Thuộc tính category\_id là thể loại của video, từng video sẽ có một thể loại nhất định. Và mỗi thể loại đều sẽ có một id của riêng nó.

Các thể loại cụ thể tương ứng với từng id như sau:

- 1 - Film & Animation
- 2 - Autos & Vehicles
- 10 - Music
- 15 - Pets & Animals
- 17 - Sports
- 18 - Short Movies
- 19 - Travel & Events
- 20 - Gaming
- 21 - Videoblogging
- 22 - People & Blogs
- 23 - Comedy
- 24 - Entertainment
- 25 - News & Politics
- 26 - Howto & Style
- 27 - Education
- 28 - Science & Technology
- 29 - Nonprofits & Activism
- 30 - Movies
- 31 - Anime/Animation
- 32 - Action/Adventure
- 33 - Classics
- 34 - Comedy

- 35 - Documentary
- 36 - Drama
- 37 - Family
- 38 - Foreign
- 39 - Horror
- 40 - Sci-Fi/Fantasy
- 41 - Thriller
- 42 - Shorts
- 43 - Shows

category\_id sẽ là biến mục tiêu (target) trong mô hình dự đoán.



Hình 5.1: Biểu đồ số lượng video theo thể loại

Với biểu đồ trên, ta có thể thấy được thể loại Music và Entertainment có số lượng video nhiều nhất. Điều này cũng phần nào phản ánh được rằng, thị hiếu của người dùng xưa này đã phân tập trung vào giải trí, âm nhạc.

Với lượng video trung bình, chúng ta có Film & Animation, Sports, People & Blogs, Comedy, News & Politics và Howto & Style. Các video có nội dung này cũng khá phổ biến, nhưng không áp đảo bằng âm nhạc và giải trí.

Còn lại là các thể loại có lượng video khá thấp, phản ánh được rằng thị hiếu người dùng không tập trung vào các thể loại như Autos & Vehicles, Travel & Events hay Nonprofits & Activism.

Để phân tích data sâu hơn, chúng ta tiến hành vẽ biểu đồ lượt likes, dislike, comment\_count để có cái nhìn tổng quan hơn. Chúng ta sẽ có code cho việc vẽ biểu đồ sau:

```

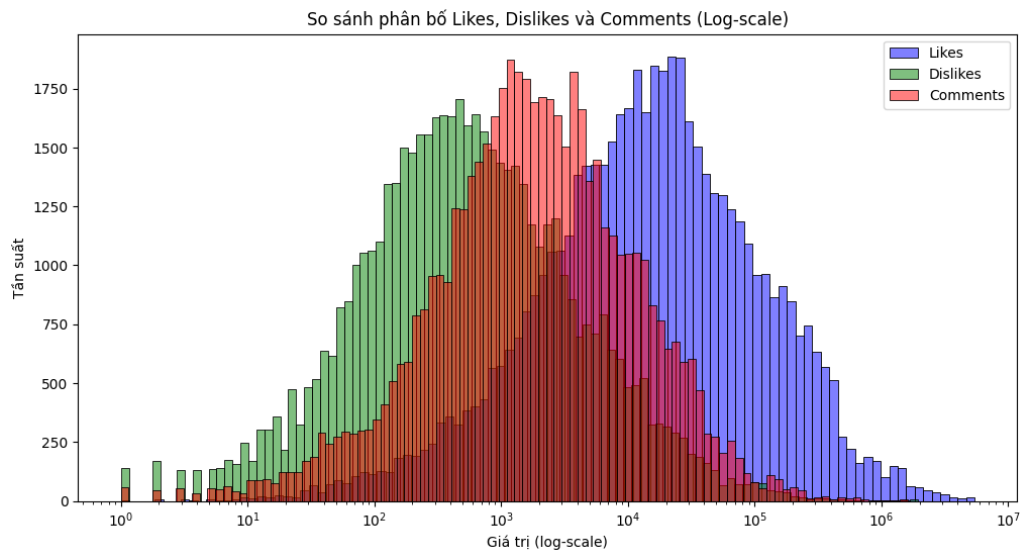
1  # Vẽ biểu đồ phân bố
2  plt.figure(figsize=(12, 6))
3
4  # Histogram số lượt thích (likes)
5  sns.histplot(data['likes'], kde=True, color='blue', bins=100, label="Likes")
6
7  # Histogram số lượt không thích (dislikes)
8  sns.histplot(data['dislikes'], kde=True, color='green', bins=100, label="Dislikes")
9
10 # Histogram số bình luận (comment_count)
11 sns.histplot(data['comment_count'], kde=True, color='red', bins=100, label="Comments")
12
13 # Thiết lập tiêu đề và nhãn
14 plt.title('So sánh phân bố Likes, Dislikes và Comments')
15 plt.xlabel('Số lượng')
16 plt.ylabel('Tần suất')
17 plt.legend()
18
19 # Hiển thị biểu đồ
20 plt.show()

```

Hình 5.2: Code vẽ biểu đồ phân bố lượng likes, dislikes, comment\_count



Sau khi đoạn code chạy thành công, ta sẽ có biểu đồ như sau:



Hình 5.3: biểu đồ so sánh phân bố likes, dislike, comment\_count

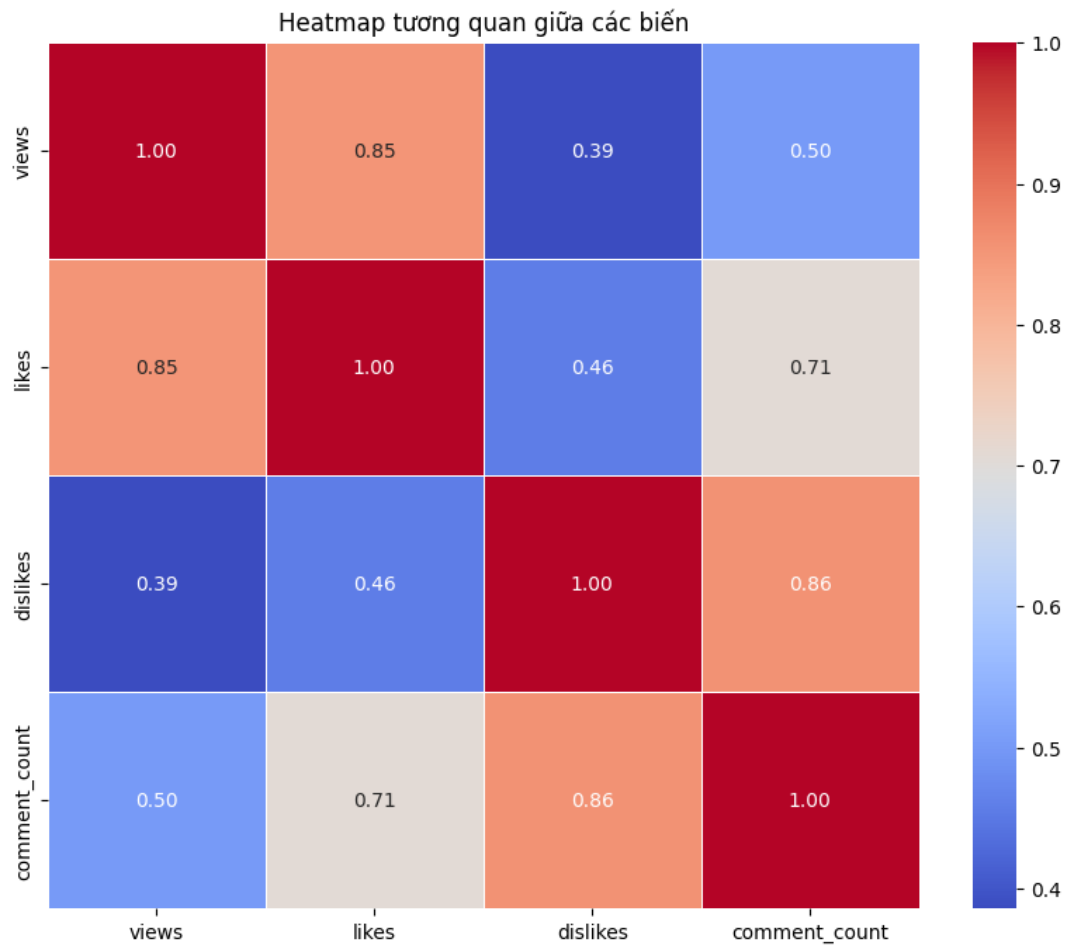
Với biểu đồ trên, ta có thể thấy được rằng, hầu hết các video chỉ nhận về số ít lượt like, dislike cũng như là comment. Các video có những chỉ số này cao thường là những video viral, trending hoặc thuộc sở hữu của các channel lớn, nổi tiếng.

Điều này cho thấy rằng, độ viral của video, hay độ viral của chủ kênh sở hữu các video đó có ảnh hưởng lớn tới lượng like, dislike và comment. Điều này cho thấy các video được đề xuất nhiều trên mạng xã hội, hay quảng cáo nhiều, sẽ thu hút nhiều lượt bình luận, lượt like hoặc dislike tùy vào nội dung của video đó.

Trong biểu đồ, ta thấy được rằng, like có số lượng lớn nhất (đường màu đỏ) với giá trị cao nhất là trên  $10^6$ , sau đó là dislike với giá trị trên  $10^5$ , cuối cùng là comment với giá trị thấp hơn dislike.

Điều này cho thấy rằng, xu hướng người xem sẽ nhấn like nhiều hơn là dislike và comment.

Chúng ta còn có biểu đồ heatmap tương quan của các biến như sau:



Hình 5.4: Heatmap tương quan giữa các biến

Trong heatmap trên, ta có thể thấy được rằng giữa biến ‘views’ và ‘likes’ có độ tương quan rất cao (0.85). Điều này có thể được hiểu rằng lượt views cao đồng nghĩa với việc lượt likes tăng theo. Điều này trên thực tế là đúng, vì các video có lượng views cao và có khả năng trending thường có lượt likes cao không kém.

Song song đó, ta còn có biến ‘likes’ và ‘comment\_count’ cũng có sự tương quan cao (0.71). ‘dislikes’ và ‘comment\_count’ cũng tương quan rất cao, điều này ngược với cặp ‘likes’ và ‘comment\_count’, nhưng thực tế vẫn đúng, vì có nhiều video trở nên viral không phải vì nó được yêu thích nhiều, mà ngược lại, video đó viral vì rất nhiều người ghét nó.

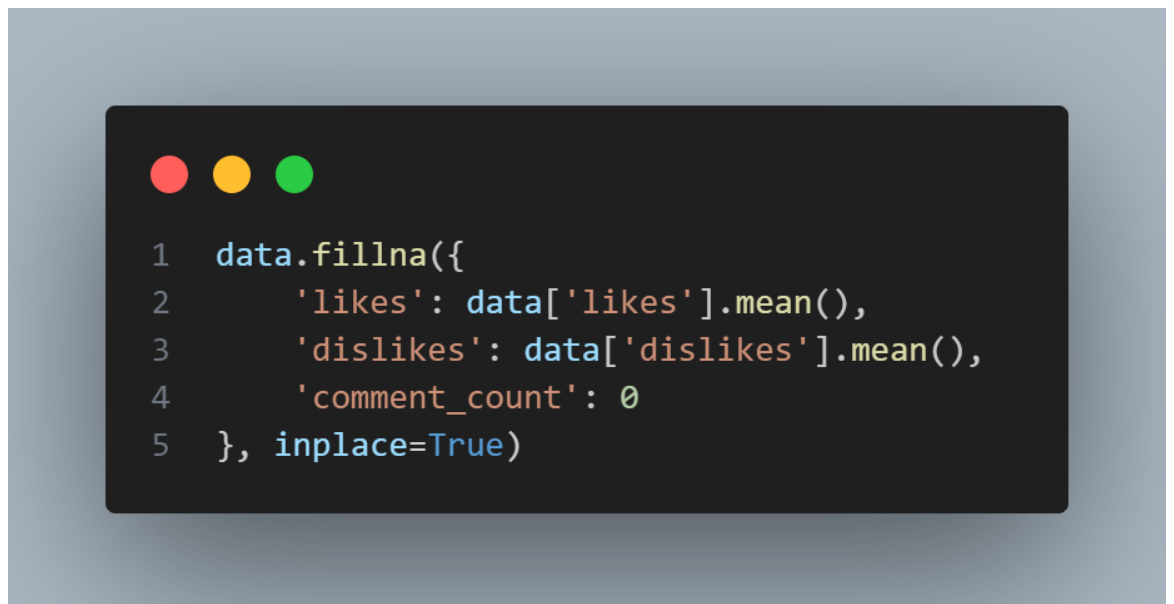
Ta còn có ‘views’ và ‘dislikes’ không tương quan quá nhiều với nhau. ‘likes’ và ‘dislike’ cũng như vậy.

### 5.1.2 Tiền xử lý dữ liệu

Sau khi đã có được data phù hợp, tiến hành kiểm tra xem data đó có bị missing hay không, có bị lỗi gì không, các biến trong data có đồng nhất với nhau hay không.

Bằng cách dùng hàm `isnull().sum()` để trả về số lượng các ô mang giá trị null hoặc NaN trong dataframe.

Sau khi kiểm tra xong, tiến hành bước cơ bản nhưng cực kỳ quan trọng đó là tiền xử lý dữ liệu bằng cách điền dữ liệu bị thiếu, bỏ hoặc sửa các giá trị không phù hợp.



Hình 5.5: Code điền dữ liệu cho các biến bị thiếu

Giá trị thiếu trong các cột số liệu (`likes`, `dislikes`, `comment_count`) có thể được điền bằng trung vị (`median`) hoặc giá trị mặc định (0).

### 5.1.3 Định nghĩa các biến và tham số của dữ liệu

Đầu tiên, cần xác định biến quan sát, đây là biến mà ta có thể đo và quan sát được một cách trực tiếp. Đây là biến chứa các đại lượng cụ thể mà mô hình cần phải dựa vào để dự đoán cho một mục tiêu nào đó.

Biến hành vi (behavior): là biến phản ánh những hành động cụ thể của con người trong một sự kiện, sự việc nào đó. Trong data này, biến hành vi sẽ là 'view'.



Hình 5.6: Biến hành vi

Biến cảm xúc (emotion): là biến phản ánh về cảm xúc, tâm lý của con người trong một sự kiện, sự việc nào đó mà data đang chứa. Trong data này, biến cảm xúc sẽ là 'likes', 'dislikes', 'comment\_count'.



Hình 5.7: Biến cảm xúc

Tiếp theo, xác định tham số tiềm ẩn. Tham số này là tham số mà ta không thể quan sát trực tiếp được, nhưng nó lại ảnh hưởng lớn tới biến quan sát. Các tham số

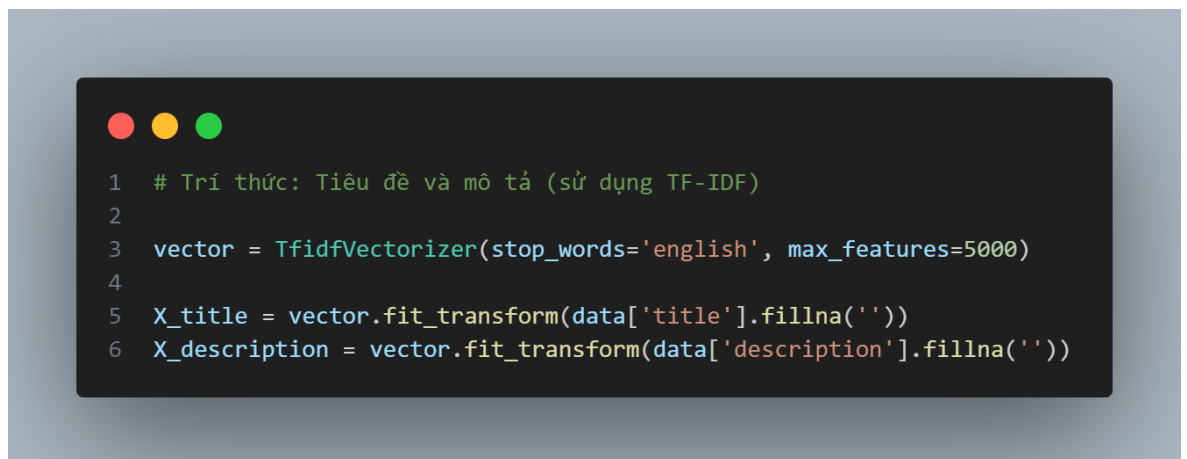
này thường liên quan tới tâm lý, các yếu tố không xác định được rõ, vô hình, khó đong đếm.

Ví dụ trong data về phản ánh sản phẩm tiêu dùng sẽ có tham số “Độ tin cậy”, đây là một tham số không thể quan sát trực tiếp, nhưng lại ảnh hưởng lớn tới hành vi người tiêu dùng, quyết định xem sản phẩm đó có được mua nhiều hay không, ...

Trong data trending video youtube, “Title” có thể là một tham số tiềm ẩn, vì nó ảnh hưởng tới mức độ trending của video, chúng ta không thể quan sát được mức độ hấp dẫn của tiêu đề vì đó là cảm quan của mỗi người.

Ta xác định được tham số tiềm ẩn sẽ là ‘title’ và ‘description’. Nhưng vì trong ‘title’ và ‘description’ chứa nhiều từ không mang quá nhiều giá trị trong việc phân tích, vì chúng xuất hiện rất thường xuyên nhưng không mang nhiều ý nghĩa như từ ‘and’, ‘of’, ...

Vì vậy, chúng ta dùng kỹ thuật trong xử lý ngôn ngữ đã nêu trước đó là TF-IDF để mô hình tập trung vào các từ cần thiết như ‘tutorial’ hay ‘trend’, ... để có thể dự đoán dễ dàng hơn. Sau đó dùng fillna(‘’) để thay thế chuỗi rỗng cho các giá trị NaN trong ‘title’ và ‘description’.



```

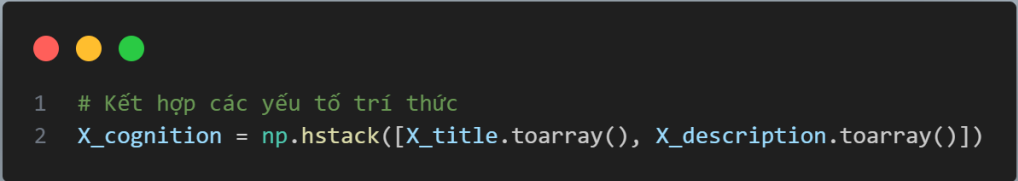
1  # Trí thức: Tiêu đề và mô tả (sử dụng TF-IDF)
2
3  vector = TfidfVectorizer(stop_words='english', max_features=5000)
4
5  X_title = vector.fit_transform(data['title'].fillna(''))
6  X_description = vector.fit_transform(data['description'].fillna(''))

```

Hình 5.8: Biến tri thức áp dụng TF-IDF

Bước định nghĩa các tham số này là rất quan trọng trong quá trình xây dựng mô hình Bayesian Mindsponge, nó giúp cho mô hình có thể phân tích và dự đoán.

Sau khi xác định các biến xong, tiến hành kết hợp các biến lại với nhau, giúp cho mô hình có thể hiểu rõ hơn các mối quan hệ của các biến bằng cách dùng hàm **hstack()** của thư viện NumPy để kết hợp các ‘features’, tạo thành đầu vào cho mô hình.

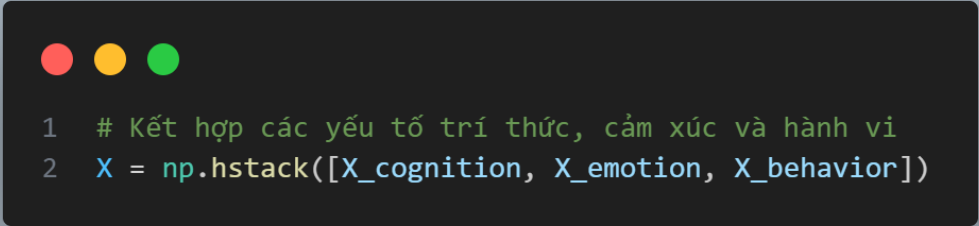


```

1  # Kết hợp các yếu tố trí thức
2  X_cognition = np.hstack([X_title.toarray(), X_description.toarray()])

```

Hình 5.9: Kết hợp các biến tri thức sau khi dùng TF-IDF



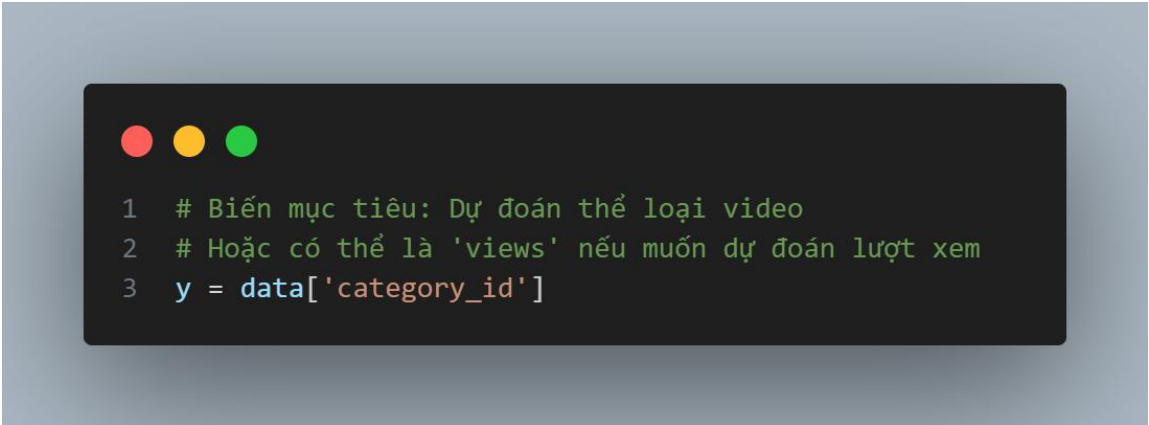
```

1  # Kết hợp các yếu tố trí thức, cảm xúc và hành vi
2  X = np.hstack([X_cognition, X_emotion, X_behavior])

```

Hình 5.10: Code kết hợp các biến tri thức, cảm xúc, hành vi lại với nhau

Sau khi kết hợp các yếu tố cần thiết lại với nhau, tiến hành xác định tham số mục tiêu, dùng để dự đoán và phân loại video sẽ thuộc thể loại nào dựa trên các đặc trưng đã xác định trước đó.



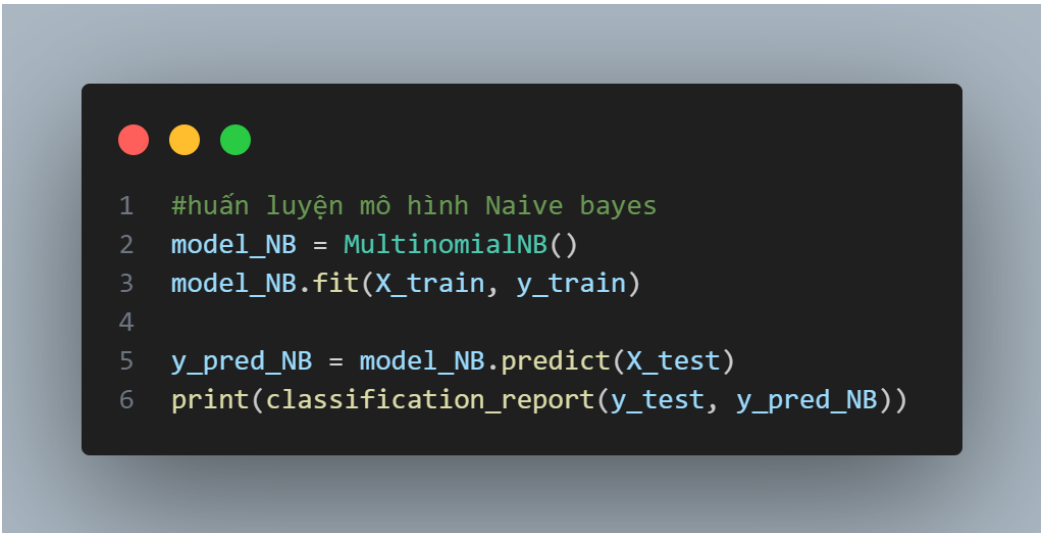
```
1 # Biến mục tiêu: Dự đoán thể loại video
2 # Hoặc có thể là 'views' nếu muốn dự đoán lượt xem
3 y = data['category_id']
```

Hình 5.11: Xác định biến mục tiêu

Như vậy, ta đã có được các giá trị cần thiết cho việc dự đoán thể loại của một video dựa theo tiêu đề, mô tả, lượng thích, không thích và số comment của một video. Bước tiếp theo cần làm đó là chọn ra một mô hình sử dụng kỹ thuật Bayesian để huấn luyện phân tích dữ liệu và đưa ra các dự đoán.

## 5.2 Áp dụng công thức Bayesian để phân tích dữ liệu

### 5.2.1 Naive Bayes



```
1 #huấn luyện mô hình Naive bayes
2 model_NB = MultinomialNB()
3 model_NB.fit(X_train, y_train)
4
5 y_pred_NB = model_NB.predict(X_test)
6 print(classification_report(y_test, y_pred_NB))
```

Hình 5.12: Dùng mô hình Naive Bayes để huấn luyện

Áp dụng mô hình Naive Bayes để thực hiện phân loại dựa trên:

- Thông tin văn bản đã xử lý từ title và description.
- Các biến số liệu như likes, dislikes, comment\_count.

Sau đó, dựa vào công thức Bayes, tính toán xác suất để video thuộc một thể loại cụ thể dựa trên các đặc trưng đầu vào.

Huấn luyện mô hình:

- Sử dụng dữ liệu huấn luyện để tính toán các xác suất tiên nghiệm và có điều kiện.
- Áp dụng mô hình Naive Bayes để phân loại video.

Đánh giá mô hình:

Sử dụng các chỉ số như độ chính xác (accuracy), độ nhạy (recall), và độ đặc hiệu (specificity) để đánh giá hiệu suất của mô hình trên tập test.

	precision	recall	f1-score	support
1	0.09	0.05	0.07	706
2	0.07	0.09	0.08	93
10	0.49	0.17	0.25	2411
15	0.04	0.15	0.07	130
17	0.07	0.06	0.06	589
19	0.00	0.16	0.01	55
20	0.10	0.09	0.09	513
22	0.12	0.01	0.02	950
23	0.11	0.31	0.16	751
24	0.21	0.01	0.03	2994
25	0.37	0.32	0.34	706
26	0.12	0.05	0.07	458
27	0.06	0.16	0.09	216
28	0.04	0.04	0.04	294
29	0.00	0.07	0.00	27
43	0.01	0.60	0.01	5
accuracy			0.11	10898
macro avg	0.12	0.15	0.09	10898
weighted avg	0.23	0.11	0.12	10898

Hình 5.13: Kết quả đánh giá mô hình Naïve Bayes



Với kết quả đánh giá trên, ta có thể thấy được rằng:

- Lớp 10 (Music) có Precision = 0.49, Recall = 0.17, F1-score = 0.25, cho thấy mô hình nhận diện lớp này khá tốt so với các lớp khác.
- Lớp 25 (News & Politics) có Precision = 0.37, Recall = 0.32, F1-score = 0.34, cũng có kết quả tốt hơn các lớp khác.
- Nhiều lớp có Recall cực thấp ( $\approx 0.01 - 0.07$ ), chứng tỏ mô hình không nhận diện tốt hầu hết các thể loại.
- Một số lớp như lớp 29, 43, 22, 24 có F1-score gần 0, nghĩa là mô hình gần như không thể phân biệt các thể loại này.

Đánh giá tổng thể mô hình:

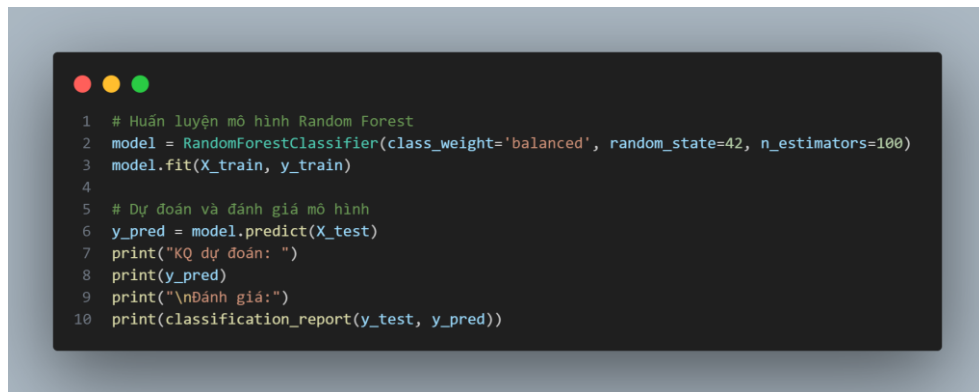
- Accuracy chỉ đạt 11%, quá thấp để sử dụng thực tế.
- Macro avg F1-score = 0.09, điều này cho thấy hiệu suất rất thấp khi xét trên toàn bộ các lớp.

Weighted avg F1-score = 0.12, phản ánh rằng mô hình chưa hoạt động tốt ngay cả khi có các lớp phổ biến hơn chi phối kết quả.

### 5.2.2 *Random Forest (Non-Bayes)*

Để có cái nhìn trực quan hơn, chúng em sử dụng một mô hình non-bayes để huấn luyện, đánh giá và so sánh với mô hình bayes.

Chúng em chọn Random Forest là mô hình sẽ được sử dụng cho việc này.



Hình 5.14: Huấn luyện Random Forest

Tổng quan đánh giá mô hình:

Đánh giá:	precision	recall	f1-score	support
1	0.99	0.96	0.97	706
2	0.99	0.71	0.82	93
10	1.00	0.99	0.99	2411
15	1.00	0.95	0.98	130
17	0.95	0.93	0.94	589
19	1.00	0.93	0.96	55
20	0.99	0.93	0.96	513
22	0.95	0.91	0.93	950
23	0.99	0.93	0.96	751
24	0.91	0.99	0.95	2994
25	0.95	0.94	0.95	706
26	0.99	0.95	0.97	458
27	1.00	0.93	0.96	216
28	0.98	0.92	0.95	294
29	1.00	0.96	0.98	27
43	1.00	1.00	1.00	5
accuracy			0.96	10898
macro avg	0.98	0.93	0.95	10898
weighted avg	0.96	0.96	0.96	10898

Hình 5.15: Đánh giá mô hình

- Accuracy = 96%, cao hơn đáng kể so với mô hình Naive Bayes trước đó (11%).
- Macro F1-score = 0.95, điều này cho thấy hiệu suất tổng thể trên tất cả các lớp rất tốt.
- Weighted F1-score = 0.96, chứng tỏ mô hình hoạt động ổn định ngay cả với các lớp có số lượng mẫu khác nhau.

Nhìn chung, mô hình không bỏ sót quá nhiều mẫu, mà nhận diện rất tốt.

### 5.3 So sánh và đánh giá

Table 5.1 So sánh kết quả đánh giá mô hình

Tiêu chí	Naive Bayes	Random Forest
Accuracy (Độ chính xác)	0.11 (11%)	0.96 (96%)
Macro Avg Precision	0.12	0.98
Macro Avg Recall	0.15	0.93
Macro Avg F1-score	0.09	0.95
Weighted Avg Precision	0.23	0.96
Weighted Avg Recall	0.11	0.96
Weighted Avg F1-score	0.12	0.96

Độ chính xác (Accuracy)

- Naive Bayes: 11% → Dự đoán sai rất nhiều.
- Random Forest: 96% → Dự đoán đúng phần lớn.

Lý do: Naive Bayes giả định các biến độc lập, nhưng dữ liệu thực tế có sự tương quan giữa các đặc trưng, khiến mô hình hoạt động kém.

Precision (Độ chính xác theo từng lớp)

- Naive Bayes: 12% (macro avg), 23% (weighted avg)
- Random Forest: 98% (macro avg), 96% (weighted avg)

Lý do: Naive Bayes có xác suất sai cao vì giả định phân phối dữ liệu theo Gaussian hoặc Multinomial, trong khi Random Forest học được mối quan hệ thực tế giữa các đặc trưng.

Recall (Khả năng nhận diện đúng của mô hình)

- Naive Bayes: 15% (macro avg), 11% (weighted avg)
- Random Forest: 93% (macro avg), 96% (weighted avg)

Lý do: Naive Bayes có xu hướng bỏ sót nhiều dữ liệu do giả định quá đơn giản. Random Forest có nhiều cây quyết định nên khả năng nhận diện đúng cao hơn.

F1-score (Trung bình giữa Precision và Recall, đánh giá tổng thể độ chính xác)

- Naive Bayes: 9% (macro avg), 12% (weighted avg)
- Random Forest: 95% (macro avg), 96% (weighted avg)

Lý do: Naive Bayes có độ chính xác thấp và bỏ sót nhiều trường hợp, dẫn đến F1-score thấp. Random Forest cân bằng tốt giữa Precision và Recall, đạt hiệu suất cao hơn nhiều.

Tuy nhiên, với hai mô hình trên, chương trình chưa thật sự áp dụng Bayesian Mindsponge.

## 5.4 Naive Bayes và Random Forest chưa phải là BMF?

### 5.4.1 Naive Bayes

Đối với Naive bayes, mô hình luôn mặc định rằng, các đặc trưng trong bộ dữ liệu luôn độc lập và không phụ thuộc vào nhau cho dù đã biết giá trị của biến mục tiêu. Điều này không đúng so với thực tế, đặc biệt là các mô hình liên quan đến nhận thức, hành vi của con người.

Trong thực tế, các đặc trưng luôn có sự tương tác và ảnh hưởng lẫn nhau. Ví dụ trong một video, lượt view có cao hay không cũng phụ thuộc vào độ viral, mà độ viral của video lại phụ thuộc vào lượt like, dislike hay comment.

Đồng thời, Naive Bayes chỉ dựa trên xác suất của từng đặc trưng mà không xét đến cách con người tiếp cận, đánh giá và cập nhật thông tin theo thời gian.

Vì vậy, ta suy ra rằng Naive Bayes chưa hẳn là BMF.

#### **5.4.2 Random Forest**

Random Forest là một phương pháp học máy phổ biến và mạnh mẽ, đặc biệt hiệu quả trong các bài toán phân loại và hồi quy. Tuy nhiên, xét về bản chất, Random Forest không phải là một mô hình Bayesian Mindsponge Framework (BMF). Điều này xuất phát từ việc Random Forest thuộc nhóm các phương pháp học máy phi Bayes (non-Bayesian), trong khi BMF dựa trên nguyên tắc suy luận Bayes để cập nhật thông tin và điều chỉnh xác suất theo dữ liệu quan sát.

Một trong những điểm khác biệt lớn nhất giữa Random Forest và BMF là khả năng giải thích kết quả. Mặc dù Random Forest có hiệu suất cao nhờ việc kết hợp nhiều cây quyết định để giảm phương sai và cải thiện độ chính xác, nhưng mô hình này thường bị xem là một "hộp đen" (black-box model). Điều này có nghĩa là mặc dù mô hình có thể đưa ra dự đoán chính xác, nhưng rất khó để hiểu rõ cách mà từng đặc trưng đầu vào đóng góp vào kết quả cuối cùng. Trong khi đó, BMF lại nhấn mạnh vào tính giải thích, cho phép người dùng hiểu rõ hơn về cách các yếu tố khác nhau ảnh hưởng đến quyết định hoặc xu hướng được mô hình hóa.

Ngoài ra, Random Forest không áp dụng Bayesian inference theo cách Mindsponge đề xuất. Bayesian inference trong BMF giúp cập nhật niềm tin về một hiện tượng theo thời gian, dựa trên dữ liệu mới, trong khi Random Forest hoạt động

theo cách tiếp cận xác suất tần suất (frequentist). Điều này có nghĩa là Random Forest không có khả năng điều chỉnh dự đoán một cách linh hoạt dựa trên thông tin tiên nghiệm và hậu nghiệm như BMF.

Tóm lại, mặc dù Random Forest là một thuật toán học máy mạnh mẽ với khả năng dự đoán chính xác, nhưng nó không phải là một mô hình BMF do thiếu tính giải thích và không dựa trên nguyên tắc suy luận Bayes. Điều này khiến cho BMF trở thành một phương pháp hấp dẫn hơn trong các lĩnh vực nghiên cứu cần sự minh bạch và khả năng hiểu rõ tác động của các yếu tố đầu vào lên kết quả phân tích

### 5.4.3 Tiếp cận BMF

BMF là một khung lý thuyết dựa trên Bayesian inference, mô tả cách con người xử lý thông tin có được từ môi trường, cập nhật niềm tin và ra quyết định. Do đó, cần mô hình hóa ba yếu tố chính:

- Nhận thức (Cognition): Mô tả cách con người tiếp nhận thông tin từ môi trường.
- Cảm xúc (Emotion): Đánh giá phản ứng và cảm xúc của con người đối với thông tin đó.
- Hành vi (Behavior): Hành động của con người dựa trên những đánh giá trước đó.

Biến số và các mối quan hệ Bayesian:

Bảng 5.2: Biến và các mối quan hệ Bayesian

Thành phần	Biến số tương ứng
Nhận thức	TF-IDF của tiêu đề, mô tả
Cảm xúc	Lượt thích, không thích, bình luận
Hành vi	Lượt xem, mức độ tương tác

Một mô hình BMF có thể được thiết lập bằng Bayesian Network, với các quan hệ xác suất:

- Nhận thức → Cảm xúc: Nội dung video ảnh hưởng đến cảm nhận của người xem.
- Cảm xúc → Hành vi: Cảm xúc tích cực hoặc tiêu cực ảnh hưởng đến lượt xem và mức độ tương tác.
- Hành vi → Thành công: Số lượt xem và tương tác ảnh hưởng đến khả năng một video trở thành xu hướng.



Hình 5.16: Lọc các biến

Đầu tiên, cần lọc ra các biến cần thiết trong việc sử dụng cho bài toán BMF. Ở đây chúng ta giữ lại cột ‘category\_id’, ‘likes’, ‘dislikes’, ‘comment\_count’ và ‘views’.

Trong pandas có một cảnh báo là ‘SettingWithCopyWarning’, đây là một cảnh báo phổ biến trong pandas khi ta cố gắng thay đổi một phần của DataFrame mà có thể là một bản sao (copy) thay vì tham chiếu (view) đến dữ liệu gốc. Vì vậy, ta sử dụng `.copy()` để không xảy ra cảnh báo này.

Sau khi lọc biến xong, tiến hành chuyển đổi dữ liệu. Mô hình Bayesian Network trong thư viện pgmpy yêu cầu dữ liệu phải dưới dạng "category". Vì vậy, cần chuyển dữ liệu cột ‘category\_id’ sang kiểu category.





```
1 df_bmf['category_id'] = df_bmf['category_id'].astype("category")
```

Hình 5.17: Chuyển đổi dữ liệu 'category\_id' thành 'category'

Sau đó, tiến hành xây dựng mô hình Bayesian Network. Để tiết kiệm bộ nhớ, cấu trúc Bayesian Network sẽ có định dạng như sau:



```
1 # định nghĩa cấu trúc mô hình phù hợp với dữ liệu
2 model_bmf = BayesianNetwork([
3     ('category_id', 'views'),
4     ('category_id', 'likes'),
5     ('likes', 'dislikes'),
6 ])
```

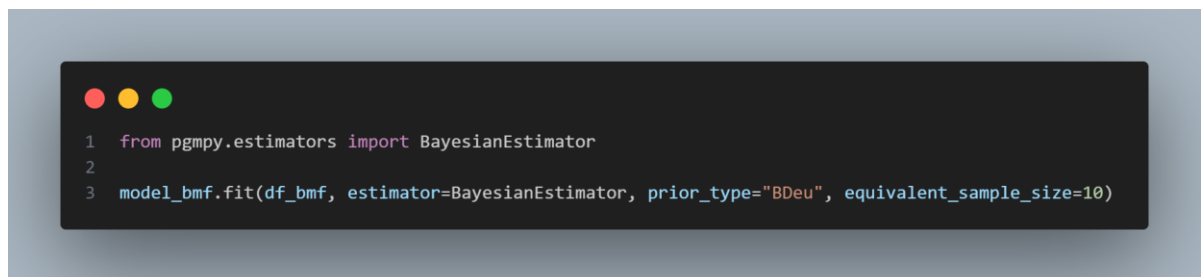
Hình 5.18: Cấu trúc mô hình phù hợp

Trong đó:

- category\_id -> views: Thể loại nội dung của video ảnh hưởng đến số lượt xem.
- category\_id -> likes: Thể loại của video cũng ảnh hưởng đến lượt thích.
- likes -> dislikes: Video nhiều likes hơn có thể tương ứng với người xem, nhưng cũng tương phản với lượt display.

Sau khi xác định cấu trúc của mô hình Bayesian Network, ta cần ước lượng các tham số để xác định xác suất có điều kiện của từng biến. Trong phần này, ta sử dụng Bayesian Estimator với prior BDeu (Bayesian Dirichlet equivalent uniform prior) và equivalent sample size = 10. Phương pháp này giúp tránh hiện tượng xác suất bằng 0 khi dữ liệu chưa quan sát đầy đủ.

Việc sử dụng BayesianEstimator với prior BDeu giúp đảm bảo rằng xác suất có điều kiện của các biến không bị quá thấp hoặc bằng 0, đặc biệt trong trường hợp dữ liệu có nhiều giá trị hiếm gặp.



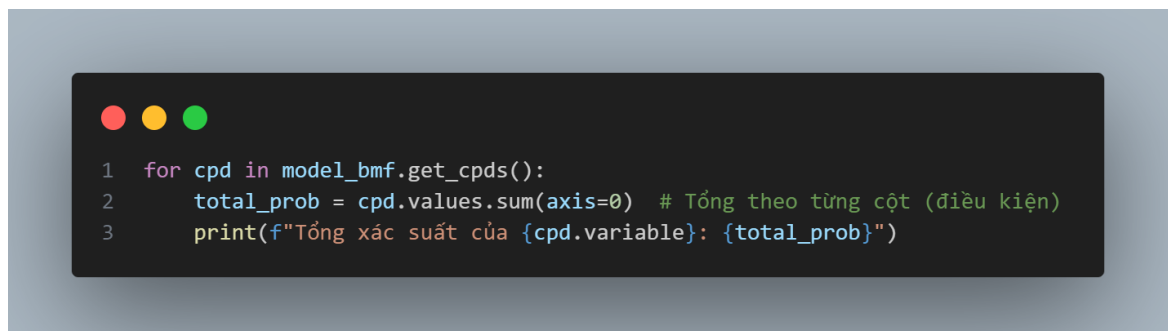
```

1 from pgmpy.estimators import BayesianEstimator
2
3 model_bmf.fit(df_bmf, estimator=BayesianEstimator, prior_type="BDeu", equivalent_sample_size=10)

```

Hình 5.19: Sử dụng Bayesian Estimator trong pgmpy

Sau khi fitting mô hình, ta cần kiểm tra tổng xác suất của từng bảng phân phối có điều kiện (CPD - Conditional Probability Distribution) để đảm bảo tính hợp lệ của mô hình. Tổng xác suất theo từng điều kiện phải bằng 1.



```

1 for cpd in model_bmf.get_cpds():
2     total_prob = cpd.values.sum(axis=0) # Tổng theo từng cột (điều kiện)
3     print(f"Tổng xác suất của {cpd.variable}: {total_prob}")

```

Hình 5.20: Kiểm tra tổng xác suất

Việc kiểm tra này giúp đảm bảo rằng mô hình Bayesian Network đã học được các xác suất hợp lệ và không có lỗi trong quá trình tính toán phân phối xác suất.

Sau khi kiểm tra xong, kết quả sẽ được hiển thị như sau:

```
Tổng xác suất của category_id: 1.0
Tổng xác suất của views: [1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
Tổng xác suất của likes: [1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
Tổng xác suất của dislikes: [1. 1. 1. ... 1. 1. 1.]
```

Hình 5.21: Kết quả kiểm tra

Với kết quả trên, ta có thể thấy được tất cả các giá trị đều bằng 1, điều này cho ta biết được rằng mô hình đã được huấn luyện đúng cách và có thể tiếp tục được sử dụng để suy luận và đánh giá.

## 5.5 Kết luận

Mô hình Bayesian Mindsponge Framework (BMF) đã được triển khai với dữ liệu YouTube để phân tích mối quan hệ giữa các biến như category\_id, views, likes, dislikes, và comment\_count. Bằng cách sử dụng Bayesian Network, mô hình đã có thể xác định các phụ thuộc xác suất giữa các biến và ước lượng các tham số bằng phương pháp Bayesian Estimator với prior BDeu.

Kết quả thu được cho thấy mô hình có thể mô tả và dự đoán sự phân bố của các biến một cách hợp lý.

### 5.5.1 Ưu điểm

Ta có thể dễ dàng mở rộng hoặc thay đổi cấu trúc của Bayesian để phù hợp với các loại dữ liệu khác nhau.

Bayesian Network cung cấp khả năng trực quan hóa, cho ta hiểu được mối quan hệ nhân quả giữa các biến trong bộ dữ liệu, điều này vô cùng hữu ích cho việc phân tích dữ liệu và đưa ra phán đoán.

### ***5.5.2 Nhược điểm***

Song song với những ưu điểm trên, Bayesian Network còn tồn tại một số hạn chế, nhược điểm nhất định. Khi số lượng biến tăng, các tham số cần học tăng, những điều này làm cho thời gian tính toán trở nên lâu hơn, nặng hơn và tốn kém tài nguyên hơn.

## CHƯƠNG 6. ĐÁNH GIÁ VÀ KẾT LUẬN

### 6.1 Điểm mạnh của BMF

Bayesian inference giúp mô hình phản ánh sự không chắc chắn của dữ liệu một cách tự nhiên thông qua phân phối xác suất. Đồng thời cho phép cập nhật kiến thức khi có dữ liệu mới, giúp mô hình ngày càng chính xác hơn.

Prior (Tiên nghiệm) trong Bayesian giúp kết hợp kiến thức nền tảng với dữ liệu quan sát, đặc biệt hữu ích khi dữ liệu không đầy đủ hoặc dữ liệu bị nhiễu. Trong Mindsponge, prior có thể đại diện cho niềm tin, định kiến, hoặc kinh nghiệm trước đây của cá nhân hoặc tổ chức nào đó.

BMF không yêu cầu lượng lớn dữ liệu đầu vào như các mô hình Machine Learning khác. Bayesian Network có thể hoạt động tốt ngay cả khi dữ liệu đầu vào bị thiếu, không đồng nhất, ...

BMF còn có thể dễ dàng mở rộng với nhiều biến số mới mà không làm mất đi tính nhất quán. BMF phù hợp với nhiều lĩnh vực khác nhau như Tâm lý học, Xã hội học, Giáo dục, Khoa học dữ liệu, ...

Bayesian Network còn cung cấp cấu trúc đồ thị, giúp dễ dàng trực quan hóa mối quan hệ giữa các biến. Các kết quả xác suất có thể được diễn giải rõ ràng, giúp hỗ trợ ra quyết định.

### 6.2 Điểm yếu của BMF

Khi số lượng biến và số trạng thái của biến tăng lên, số lượng tham số cần tính toán tăng theo cấp số nhân. Điều này làm cho quá trình tính toán trở nên chậm và tiêu tốn rất nhiều tài nguyên nếu không có chiến lược tối ưu hóa cụ thể.

Tuy không yêu cầu lượng lớn dữ liệu đầu vào, BMF lại phụ thuộc vào chất lượng của bộ dữ liệu đó. Nếu không tiền xử lý dữ liệu tốt, mô hình có thể bị overfitting. Hoặc nếu dữ liệu đầu vào có quá nhiều giá trị bị thiếu, việc học của mô hình sẽ trở nên vô cùng khó khăn.

Cùng với đó, nếu cấu trúc mạng không phản ánh đúng mối quan hệ thực tế, kết quả phân tích sẽ không chính xác.

Sau cùng, Bayesian Network chỉ mô hình hóa được quan hệ xác suất có điều kiện, không thể xử lý tốt được các quan hệ phi tuyến tính phức tạp.

### 6.3 Khắc phục

Để khắc phục cho việc cần chi phí cao để mô hình có thể tính toán được với lượng biến lớn, ta cần giảm số lượng biến không cần thiết đi.

Sử dụng các kỹ thuật tiền xử lý dữ liệu để có một dữ liệu thật tốt cho mô hình không bị overfitting trong quá trình học.

Cuối cùng, chúng ta có thể kết hợp Bayesian Network với các phương pháp khác như Decision Tree để khai thác tốt hơn các mối quan hệ phi tuyến.

### 6.4 Tổng kết

Bayesian Mindsponge Framework (BMF) là một phương pháp tiên tiến trong mô hình hóa và phân tích dữ liệu, đặc biệt hữu ích trong các lĩnh vực như khoa học xã hội, tâm lý học, kinh tế, giáo dục và phân tích dữ liệu lớn, phân tích hành vi, cảm xúc con người. BMF kết hợp nguyên lý của tư duy Bayesian với mô hình Mindsponge, giúp xử lý thông tin hiệu quả và cập nhật xác suất linh hoạt dựa trên dữ liệu quan sát. Nhờ đó, phương pháp này có thể hỗ trợ các nhà nghiên cứu trong việc khám phá các yếu tố tác động đến hành vi, dự đoán xu hướng và giải thích những biến động trong dữ liệu.

Một trong những lợi thế nổi bật của BMF là khả năng tiếp cận dữ liệu theo hướng xác suất, giúp mô hình hóa sự không chắc chắn trong hệ thống và đưa ra các dự đoán có độ tin cậy cao. Không giống như các phương pháp truyền thống thường chỉ đưa ra kết quả cố định dựa trên dữ liệu huấn luyện, BMF có thể điều chỉnh và cập nhật các dự đoán khi dữ liệu có sự thay đổi. Điều này đặc biệt quan trọng trong

các lĩnh vực mà dữ liệu có tính biến động cao, chẳng hạn như thị trường tài chính, nghiên cứu hành vi người tiêu dùng, hoặc các hiện tượng xã hội phức tạp.

Tuy nhiên, giống như bất kỳ phương pháp phân tích dữ liệu nào, BMF cũng có những hạn chế nhất định cần được xem xét cẩn thận. Một trong những thách thức lớn nhất là yêu cầu cao về khả năng tính toán. Việc ước lượng tham số trong mô hình Bayesian thường đòi hỏi các thuật toán phức tạp như Markov Chain Monte Carlo (MCMC) hoặc Variational Inference, vốn có thể tiêu tốn nhiều tài nguyên máy tính và thời gian xử lý. Khi làm việc với các tập dữ liệu lớn và có nhiều biến số, việc tối ưu hóa mô hình và đảm bảo tốc độ tính toán trở thành một vấn đề quan trọng mà người sử dụng cần phải cân nhắc.

Ngoài ra, việc lựa chọn phân phối tiên nghiệm phù hợp là một yếu tố quan trọng ảnh hưởng đến hiệu suất của BMF. Phân phối tiên nghiệm đóng vai trò như một giả định ban đầu về tham số cần ước lượng, và nếu được chọn không chính xác, nó có thể làm sai lệch kết quả phân tích. Điều này đòi hỏi người sử dụng không chỉ có kiến thức về xác suất thống kê mà còn phải hiểu rõ bản chất của dữ liệu và vấn đề nghiên cứu. Trong một số trường hợp, việc thử nghiệm nhiều phân phối tiên nghiệm khác nhau và đánh giá tác động của chúng lên mô hình là điều cần thiết để đảm bảo kết quả đáng tin cậy.

Bên cạnh những thách thức về tính toán và xác định tiên nghiệm, chất lượng dữ liệu đầu vào cũng ảnh hưởng lớn đến hiệu quả của mô hình BMF. Nếu dữ liệu không đầy đủ, chứa nhiều giá trị nhiễu hoặc có sai sót trong quá trình thu thập, mô hình có thể tạo ra kết quả không chính xác hoặc mang tính thiên lệch. Do đó, bước tiền xử lý dữ liệu, bao gồm việc loại bỏ giá trị ngoại lệ, xử lý dữ liệu bị thiếu và đảm bảo tính đồng nhất trong bộ dữ liệu, là vô cùng quan trọng. Việc áp dụng các kỹ thuật như kiểm tra dữ liệu đầu vào, trực quan hóa phân phối dữ liệu và sử dụng các phương pháp thống kê để phát hiện lỗi có thể giúp cải thiện độ tin cậy của mô hình.

Dù còn một số hạn chế nhất định, BMF vẫn là một công cụ mạnh mẽ nếu được sử dụng đúng cách. Với sự hỗ trợ của các công cụ lập trình hiện đại như

Python và R, người dùng có thể tận dụng các thư viện chuyên biệt như PyMC, TensorFlow Probability hoặc Stan để xây dựng và tối ưu hóa mô hình Bayesian một cách hiệu quả. Các thư viện này không chỉ giúp giảm bớt khối lượng công việc tính toán mà còn cung cấp nhiều công cụ hỗ trợ kiểm định và đánh giá mô hình, giúp người dùng nhanh chóng điều chỉnh và cải thiện hiệu suất phân tích.

Một trong những ứng dụng tiềm năng của BMF là trong phân tích hành vi con người, nơi mô hình có thể được sử dụng để tìm hiểu cách con người tiếp nhận và xử lý thông tin. Trong lĩnh vực tâm lý học, BMF có thể giúp nghiên cứu về nhận thức, ra quyết định và tác động của môi trường xã hội đến hành vi cá nhân. Tương tự, trong kinh tế học, mô hình này có thể hỗ trợ phân tích hành vi tiêu dùng, dự đoán xu hướng thị trường và tối ưu hóa chiến lược kinh doanh. Trong giáo dục, BMF có thể được sử dụng để nghiên cứu về phương pháp giảng dạy, đo lường mức độ tiếp thu kiến thức của học sinh và đánh giá hiệu quả của các chương trình đào tạo.

Nhìn chung, Bayesian Mindsponge Framework là một phương pháp có nhiều tiềm năng trong việc phân tích và dự đoán dữ liệu, đặc biệt là trong các hệ thống phức tạp và có sự không chắc chắn cao. Mặc dù tồn tại một số thách thức trong tính toán, xử lý dữ liệu và lựa chọn phân phối tiên nghiệm, nhưng với sự phát triển của công nghệ tính toán và các công cụ hỗ trợ lập trình, BMF có thể trở thành một công cụ đắc lực cho các nhà nghiên cứu và chuyên gia trong nhiều lĩnh vực. Bằng cách áp dụng BMF một cách khoa học và linh hoạt, chúng ta có thể khai thác tối đa sức mạnh của phương pháp này để hiểu rõ hơn về thế giới xung quanh và đưa ra những quyết định chính xác hơn dựa trên dữ liệu.



## TÀI LIỆU THAM KHẢO

### Tiếng Anh

[1] Minh Hoang Nguyen, Tam Tri Le, Quy Khuc (2022) Bayesian Mindsponge Framwork

[2] Viet Phuong La, Tam Tri Le (2022) Introduction to Bayesian Mindsponge Framework analytics: an innovative method for social and psychological research

### Tiếng Việt

[3] Hoàng Vui (2023) TF-IDF là gì? Sử dụng TF-IDF tối ưu quy trình SEO

[4] Codelearn.io - Python cơ bản

[5] Aptech (2021) Ứng dụng của Python phổ biến nhất trong thực tế

[6] Topdev.vn (2024) Python là gì? Tổng hợp kiến thức cho người mới bắt đầu

[7] Nguyễn Tấn Vũ (2019) Đôi nét về TF-IDF trong xử lý ngôn ngữ tự nhiên.

[8] Dương Phạm (2016) TF-IDF

[9] Nguyễn Văn Hiếu (2024) TF-IDF là gì?

[10] Wikipedia (2023) Cơ chế mindsponge