BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

BÙI THỊ THANH XUÂN

MỘT SỐ PHƯƠNG PHÁP NGẪU NHIÊN CHO BÀI TOÁN CỰC ĐẠI HÓA XÁC SUẤT HẬU NGHIỆM KHÔNG LỒI TRONG HỌC MÁY

LUẬN ÁN TIẾN SĨ HỆ THỐNG THÔNG TIN

HÀ NỘI–2020

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

BÙI THỊ THANH XUÂN

MỘT SỐ PHƯƠNG PHÁP NGẪU NHIÊN CHO BÀI TOÁN CỰC ĐẠI HÓA XÁC SUẤT HẬU NGHIỆM KHÔNG LỒI TRONG HỌC MÁY

Ngành: Hệ thống thông tin
Mã số: 9480104

LUẬN ÁN TIẾN SĨ HỆ THỐNG THÔNG TIN

TẬP THỂ HƯỚNG DẪN KHOA HỌC:

1. PGS.TS. THÂN QUANG KHOÁT
2. TS. NGUYỄN THỊ OANH

HÀ NỘI–2020

LỜI CAM ĐOAN

Tôi xin cam đoan các kết quả trình bày trong luận án là công trình nghiên cứu của bản thân nghiên cứu sinh trong thời gian học tập và nghiên cứu tại Đại học Bách khoa Hà Nội dưới sự hướng dẫn của tập thể hướng dẫn khoa học. Các số liệu, kết quả trình bày trong luận án là hoàn toàn trung thực. Các kết quả sử dụng tham khảo đều đã được trích dẫn đầy đủ và theo đúng quy định.

Hà Nội, ngày tháng 02 năm 2020

Nghiên cứu sinh

Bùi Thị Thanh Xuân

TẬP THỂ HƯỚNG DẪN KHOA HỌC

LỜI CẢM ƠN

Trong quá trình nghiên cứu và hoàn thành luận án này, nghiên cứu sinh đã nhận được nhiều sự giúp đỡ và đóng góp quý báu. Đầu tiên, nghiên cứu sinh xin được bày tỏ lòng biết ơn sâu sắc tới tập thể hướng dẫn: PGS.TS. Thân Quang Khoát và TS. Nguyễn Thị Oanh. Các thầy cô đã tận tình hướng dẫn, giúp đỡ nghiên cứu sinh trong suốt quá trình nghiên cứu và hoàn thành luận án. Nghiên cứu sinh xin chân thành cảm ơn Bộ môn Hệ thống thông tin và Phòng thí nghiệm Khoa học dữ liệu, Viện Công nghệ thông tin và truyền thông - Trường Đại học Bách khoa Hà Nội, nơi nghiên cứu sinh học tập đã tạo điều kiện, cho phép nghiên cứu sinh có thể tham gia nghiên cứu trong suốt thời gian học tập. Nghiên cứu sinh xin chân thành cảm ơn Phòng Đào tạo - Trường Đại học Bách Khoa Hà Nội đã tạo điều kiện để nghiên cứu sinh có thể hoàn thành các thủ tục bảo vệ luận án tiến sĩ. Cuối cùng, nghiên cứu sinh xin gửi lời cảm ơn sâu sắc tới gia đình, bạn bè đồng nghiệp đã luôn động viên, giúp đỡ nghiên cứu sinh vượt qua khó khăn để đạt được những kết quả nghiên cứu như hôm nay.

MỤC LỤC

DANH MỤC CÁC TỪ VIẾT TẮT VÀ THUẬT NGỮ

Viết tắt	**Tiếng Anh**	**Tiếng Việt**
BOPE	Bernoulli randomness in OPE	Phương pháp BOPE
CCCP	Concave-Convex Procedure	Phương pháp CCCP
CGS	Collapsed Gibbs Sampling	Phương pháp CGS
CTMP	Collaborative Topic Model for Poisson	Mô hình CTMP
CVB	Collapsed Variational Bayes	Phương pháp CVB
CVB0	Zero-order Collapsed Variational Bayes	Phương pháp CVB0
DC	Difference of Convex functions	Hiệu của hai hàm lồi
DCA	Difference of Convex Algorithm	Thuật toán DCA
EM	Expectation–Maximization algorithm	Thuật toán tối đa hóa kì vọng
ERM	Empirical risk minimization	Cực tiểu hóa hàm rủi ro thực nghiệm
FW	Frank-Wolfe	Thuật toán tối ưu Frank-Wolfe
GD	Gradient Descent	Thuật toán tối ưu GD
GOA	Graduated Optimization Algorithm	Thuật toán GOA
GOPE	Generalized Online Maximum a Posteriori Estimation	Phương pháp GOPE
GradOpt	Graduated Optimization	Phương pháp tối ưu GradOpt
GS	Gibbs Sampling	Phương pháp lấy mẫu Gibbs
HAMCMC	Hessian Approximated MCMC	Phương pháp tối ưu HAMCMC
LDA	Latent Dirichlet Allocation	Mô hình chủ đề ẩn
LIL	Law of the Iterated Logarithm	Luật logarit lặp
LPP	Log Predictive Probability	Độ đo LPP
LSA	Latent Semantic Analysis	Phân tích ngữ nghĩa ẩn
LSI	Latent Semantic Indexing	Chỉ mục ngữ nghĩa ẩn
MAP	Maximum a Posteriori Estimation	Phương pháp cực đại hóa ước lượng xác suất hậu nghiệm
MCMC	Markov Chain Monte Carlo	Phương pháp Monte Carlo
MLE	Maximum Likelihood Estimation	Ước lượng hợp lý cực đại
NPMI	Normalised Pointwise Mutual Information	Độ đo NPMI

Viết tắt	**Tiếng Anh**	**Tiếng Việt**
OFW	Online Frank-Wolfe algorithm	Thuật toán tối ưu Online Frank-Wolfe
OPE	Online maximum a Posteriori Estimation	Cực đại hóa ước lượng hậu nghiệm ngẫu nhiên
PLSA	Probabilistic Latent Semantic Analysis	Phân tích ngữ nghĩa ẩn xác suất
pLSI	probabilistic Latent Semantic Indexing	Chỉ mục ngữ nghĩa ẩn xác suất
PMD	Particle Mirror Decent	Phương pháp tối ưu PMD
Prox-SVRG	Proximal SVRG	Phương pháp Prox-SVRG
SCSG	Stochastically Controlled Stochastic Gradient	Phương pháp SCSG
SGD	Stochastic Gradient Descent	Thuật toán giảm gradient ngẫu nhiên
SMM	Stochastic Majorization-Minimization	Phương pháp SMM
SVD	Single Value Decomposition	Phân tích giá trị riêng
SVRG	Stochastic Variance Reduced Gradient	Phương pháp SVRG
TM	Topic Models	Mô hình chủ đề
VB	Variational Bayes	Phương pháp biến phân Bayes
VE	Variable Elimination	Phương pháp VE
VI	Variational Inference	Suy diễn biến phân

DANH MỤC HÌNH VẼ

DANH MỤC BẢNG

DANH MỤC KÝ HIỆU TOÁN HỌC

Ký hiệu	Ý nghĩa
x, y, N, k	In nghiêng, chữ thường hoặc hoa, là các số vô hướng
$\mathbf{x}$, $\mathbf{y}$	In đậm, chữ thường, là các véc-tơ
x_i	Phần tử thứ i của véc tơ $\mathbf{x}$
$\mathbf{A}$, $\mathbf{B}$	In đậm, chữ hoa, là các ma trận
$\mathbf{A}^T$	chuyển vị của ma trận $\mathbf{A}$
$\mathbf{A}^{-1}$	Ma trận nghịch đảo của ma trận vuông $\mathbf{A}$
$\|\mathbf{x}\|$	Chuẩn của véc tơ $\mathbf{x}$
$E(X)$	Kỳ vọng của biến ngẫu nhiên X
$D(X)$	Phương sai của biến ngẫu nhiên X
$B(n, p)$	Phân phối nhị thức với tham số n và p
$N(\mu, \sigma^2)$	Phân phối chuẩn với tham số μ và σ
$\mathbb{R}$	Tập hợp các số thực
$\mathbb{N}$	Tập hợp các số tự nhiên
$\mathbb{R}^n$	Không gian véc tơ n chiều
$\in$	Thuộc về
∇f	Gradient của hàm f
$\forall x$	Với mọi x
$\log(x)$	*logarit* tự nhiên của số thực dương x
$\exp(x)$	Hàm mũ e^x

MỞ ĐẦU

1. Bối cảnh nghiên cứu

Nghiên cứu về học máy, nghiên cứu sinh nhận thấy quá trình giải một bài toán trong học máy thường gồm ba bước chính: bước mô hình hóa, bước học và bước suy diễn. Trong đó, *mô hình hóa* là tìm một mô hình thích hợp cho bài toán cần giải quyết, *học* là quá trình tối ưu các tham số của mô hình và *suy diễn* là bước dự đoán kết quả đầu ra của mô hình dựa trên các tham số đã huấn luyện. Ký hiệu $\boldsymbol{x}$ là tập các tham số của mô hình, khi đó bước học chính là quá trình ước lượng tham số, tức là tìm tham số $\boldsymbol{x}$ sao cho dữ liệu sẵn có và mô hình khớp với nhau nhất. Việc tối ưu tham số, hay còn gọi là quá trình học tham số, là ý tưởng chính của các bài toán học máy nhằm tìm được mối tương quan giữa các đầu vào và đầu ra dựa trên dữ liệu huấn luyện. Một phương pháp ước lượng tham số thông dụng được sử dụng trong học máy thống kê chính là phương pháp ước lượng hợp lý cực đại MLE (Maximum Likelihood Estimation) [1, 2]. MLE thực hiện chủ yếu dựa trên các dữ liệu quan sát và thường làm việc tốt trên các mô hình có dữ liệu huấn luyện đủ lớn [3, 4, 5, 6]. Giả sử $\boldsymbol{x}$ là tập các tham số của mô hình và D là tập dữ liệu quan sát, khi đó ước lượng MLE chính là quá trình tối ưu tham số $\boldsymbol{x}$ theo xác suất:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} P(D|\boldsymbol{x}) \tag{0.1}$$

trong đó xác suất $P(D|\boldsymbol{x})$ được gọi là *likelihood* của tham số $\boldsymbol{x}$. Phương pháp MLE được xây dựng dựa trên hàm likelihood và tìm kiếm giá trị tối ưu của $\boldsymbol{x}$ để xác suất $P(D|\boldsymbol{x})$ đạt cực đại. Như đã đề cập, MLE chính là tìm cách giải thích hợp lý cho các dữ liệu quan sát được. Do xác suất $P(D|\boldsymbol{x})$ thường nhỏ, để tránh sai số tính toán, người ta thường dùng logarit tự nhiên của hàm likelihood để đưa hàm mục tiêu về dạng thuận tiện hơn. Khi đó, bài toán MLE đưa về dạng sau:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} \log P(D|\boldsymbol{x}) \tag{0.2}$$

Nếu chúng ta xem xét bài toán MLE (0.1) dưới góc độ của bài toán tối ưu với hàm mục tiêu $P(D|\boldsymbol{x})$ thì bài toán MLE (0.1) có thể được giải bằng các phương pháp tối ưu thông dụng như phương pháp nhân tử Lagrange [7],

Gradient Descent (GD) [8], Stochastic Gradient Descent (SGD) [8, 9] hay bằng phương pháp Expectation-Maximization (EM) [2, 10, 11]. Tuy nhiên, phương pháp MLE được biết đến với xu hướng phù hợp với dữ liệu, nên hiện tượng quá khớp có thể trở nên nghiêm trọng hơn đối với các mô hình phức tạp liên quan đến dữ liệu trong thế giới thực với số chiều lớn như dữ liệu hình ảnh, tiếng nói và văn bản. MLE thường làm việc không hiệu quả trong trường hợp có quá ít dữ liệu huấn luyện [12, 13, 14]. Ngoài ra, việc cực đại hóa hàm likelihood của MLE là không dễ dàng khi đạo hàm của nó là khó giải, cũng như không phải lúc nào cũng có thể giải được MLE trực tiếp bằng các phương pháp tích phân giải tích.

Khắc phục nhược điểm của MLE, chúng ta có thể ước lượng tham số mô hình theo một cách tiếp cận khác, đó là sử dụng phương pháp cực đại hóa ước lượng xác suất hậu nghiệm MAP (Maximum A Posteriori Estimation) [15]. Khác với MLE, phương pháp MAP không những dựa trên dữ liệu huấn luyện mà còn dựa trên những thông tin đã biết của tham số. Ước lượng MAP chính là tối ưu tham số $\boldsymbol{x}$ theo xác suất có điều kiện:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} \underbrace{P(\boldsymbol{x}|D)}_{\text{Posterior}} \tag{0.3}$$

trong đó xác suất $P(\boldsymbol{x}|D)$ được gọi là *xác suất hậu nghiệm (posterior probability)* của tham số $\boldsymbol{x}$. Thông thường, hàm tối ưu trong (0.3) rất khó xác định trực tiếp [16, 17]. Vì vậy, để giải bài toán MAP, chúng ta thường sử dụng quy tắc Bayes

$$P(\boldsymbol{x}|D) = \frac{P(D|\boldsymbol{x}) \times P(\boldsymbol{x})}{P(D)} \propto P(D|\boldsymbol{x}) \times P(\boldsymbol{x})$$

và đưa bài toán MAP (0.3) về dạng:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} [P(D|\boldsymbol{x}) \times P(\boldsymbol{x})] \tag{0.4}$$

trong đó xác suất $P(\boldsymbol{x})$ gọi là *xác suất tiên nghiệm (prior)* của tham số $\boldsymbol{x}$. Theo công thức (0.4) thấy rằng xác suất hậu nghiệm $P(\boldsymbol{x}|D)$ tỉ lệ thuận với tích của thành phần likelihood $P(D|\boldsymbol{x})$ và prior $P(\boldsymbol{x})$ và khi $P(\boldsymbol{x})$ là prior liên hợp thì bài toán MAP (0.4) trở nên dễ giải hơn [18]. Như vậy, việc chọn prior phù hợp giúp cho việc tối ưu bài toán MAP được thuận lợi hơn. Trong một số trường hợp, hàm mục tiêu của (0.4) khá nhỏ, sai số tính toán có thể xảy ra. Tận dụng tính chất đơn điệu tăng của hàm logarit, người ta thường lấy logarit hàm mục tiêu của (0.4) và viết lại bài toán MAP (0.4) dưới dạng:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} [\log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})] \tag{0.5}$$

Như vậy, điểm khác biệt lớn của MAP so với MLE là hàm mục tiêu của MAP có thêm thành phần phân phối tiên nghiệm $P(\boldsymbol{x})$ của $\boldsymbol{x}$. Phân phối này chính là những thông tin ta biết trước về $\boldsymbol{x}$. Thông qua (0.5), thấy rằng MAP có vai trò là kỹ thuật hiệu chỉnh của phương pháp MLE với $\log P(D|\boldsymbol{x})$ là phần hàm chính, $\log P(\boldsymbol{x})$ là phần hiệu chỉnh. Theo quan điểm của suy diễn Bayes, MLE là một trường hợp đặc biệt của MAP [19]. MAP là một phương pháp có khả năng giúp mô hình tránh hiện tượng quá khớp, đặc biệt MAP thường mang lại hiệu quả cao hơn MLE trong trường hợp có ít dữ liệu huấn luyện.

Ước lượng MAP có vai trò quan trọng trong nhiều mô hình thống kê với các biến ẩn hay các tham số không chắc chắn. Có rất nhiều nghiên cứu liên quan đến ước lượng MAP [20, 21, 22, 23, 24] hay ứng dụng của MAP vào các bài toán ngược của Bayes vô hạn [25], xử lý ảnh [26, 27], phân tích văn bản [28, 29, 30], thậm chí trong vật lý lượng tử [24]. Theo hiểu biết của nghiên cứu sinh, ước lượng MAP được sử dụng nhiều trong mô hình đồ thị xác suất [31, 16, 14, 17]. Có nhiều cách tiếp cận để giải bài toán MAP như suy diễn biến phân [32, 33] hay phương pháp lấy mẫu MCMC [34, 35],... Một hướng tiếp cận khác là xem xét bài toán MAP (0.5) dưới góc nhìn của bài toán tối ưu toán học:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}}[f(\boldsymbol{x}) = \log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})] \tag{0.6}$$

trong đó hàm mục tiêu có dạng $f(\boldsymbol{x}) = \log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})$. Khi đó có thể áp dụng các phương pháp tối ưu ngẫu nhiên để giải chúng [36]. Trong một số trường hợp bài toán MAP có thể được giải hiệu quả bằng các phương pháp tối ưu lồi ngay cả ở trong trường hợp số chiều lớn [8, 27]. Mức độ khó giải của bài toán (0.6) phụ thuộc vào đặc điểm của hàm mục tiêu $f(\boldsymbol{x})$. Trong thực tế, khi làm việc với các mô hình học máy thống kê, hàm mục tiêu $f(\boldsymbol{x})$ thường rất phức tạp, khó phân tích và thường là hàm không lồi có thể tốn kém về mặt tính toán khi đánh giá [28, 37, 38].

Bài toán MAP không lồi thường hay xuất hiện gắn liền với các mô hình học máy làm việc với dữ liệu lớn nên các phương pháp giải đúng thường không khả thi. Vì vậy một hướng tiếp cận phổ biến và hiệu quả hơn cho bài toán MAP không lồi này chính là các phương pháp xấp xỉ. Theo tìm hiểu, một số phương pháp xấp xỉ như phương pháp Variational Bayes (VB) [39], collapsed Variational Bayes (CVB) [40, 41], CVB0 [42], Collapsed Gibbs Sampling (CGS) [43], Concave-Convex procedure (CCCP) [44], Stochastic Majorization-Minimization (SMM) [45], Frank-Wolfe (FW) [46], Online-FW [47] hay Block-coordinate Frank-Wolfe

[48] có thể được áp dụng để giải bài toán ước lượng hậu nghiệm. Ngoài ra, phương pháp Particle Mirror Decent (PMD) [49] và HAMCMC [50] cũng đã được đề xuất cho bài toán ước lượng phân phối hậu nghiệm đầy đủ. Các phương pháp đề cập có thể coi là các phương pháp suy diễn tiên tiến. Tuy nhiên khi nghiên cứu và phân tích đặc điểm của chúng, nhận thấy trong các phương pháp đề cập vẫn còn một số nhược điểm tồn tại. Ví dụ, một số phương pháp đã nêu chỉ áp dụng được cho một mô hình cụ thể hoặc chúng chưa đáp ứng được các tiêu chuẩn quan trọng như sự hội tụ, tốc độ hội tụ, tính linh hoạt hay tính hiệu chỉnh. Chúng tôi chưa nhìn thấy bất kỳ phân tích lý thuyết nào về khả năng suy diễn nhanh của các phương pháp như VB, CVB, CVB0 và CGS. Mặc dù phương pháp CCCP và SMM đảm bảo hội tụ đến một điểm dừng của bài toán suy diễn, tuy nhiên tốc độ hội tụ của CCCP và SMM chưa được xác định đối với bài toán không lồi tổng quát [44, 45]. FW là một phương pháp tổng quát giải bài toán tối ưu lồi. [51] và [52] đã chỉ ra rằng thuật toán FW có thể được sử dụng hiệu quả để suy diễn cho các mô hình chủ đề. OFW là một biến thể ngẫu nhiên của FW cho các bài toán lồi. Một đặc điểm quan trọng của FW và OFW chính là chúng có thể hội tụ nhanh và cho nghiệm thưa. Tuy nhiên, hạn chế của chúng là chỉ áp dụng cho các bài toán lồi, chưa đáp ứng cho các mô hình không lồi trong học máy. Thuật toán PMD [49] và HAMCMC [50] đều dựa trên lấy mẫu để ước lượng phân phối xác suất hậu nghiệm, trong đó PMC có tốc độ hội tụ $\mathcal{O}(T^{-1/2})$ trong khi HAMCMC có tốc độ hội tụ $\mathcal{O}(T^{-1/3})$ với T là số bước lặp của thuật toán. Thuật toán Online Maximum a Posteriori Estimation (OPE) [28] đã được đề xuất để giải bài toán MAP trong các mô hình đồ thị xác suất với tốc độ hội tụ là $\mathcal{O}(1/T)$. OPE là một thuật toán tối ưu ngẫu nhiên được cải tiến từ thuật toán OFW [47] để giải bài toán MAP không lồi và có tốc độ hội tụ nhanh vượt qua nhiều thuật toán ngẫu nhiên hiện có khi giải bài toán MAP không lồi.

Mặc dù ước lượng MAP có nhiều ưu thế so với MLE trên phương diện có thể làm việc với dữ liệu huấn luyện ít, có khả năng hiệu chỉnh, tuy nhiên, tìm đến các phương pháp hiệu quả giải bài toán MAP là việc khó khăn. Và nguyên nhân chính dẫn đến khó khăn của bài toán MAP nằm ở chỗ hàm mục tiêu $f(\boldsymbol{x}) = \log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})$ trong nhiều trường hợp là hàm không lồi, khó tìm được cực đại, dẫn đến giải trực tiếp bài toán MAP không khả thi [37]. Chúng ta phải đối mặt với thách thức lớn: *Làm thế nào để giải hiệu quả bài toán MAP trong các mô hình đồ thị xác suất khi hàm mục tiêu là không lồi?* Khi đó, bài

toán MAP (0.6) có thể là không khả thi. Do vậy, đề xuất ra các thuật toán hiệu quả đảm bảo về lý thuyết và thực nghiệm để giải bài toán MAP không lồi thu hút sự quan tâm đồng thời cũng là thách thức của học máy thống kê.

2. Động lực thúc đẩy

Từ bối cảnh nghiên cứu đã được phân tích ở trên, nghiên cứu sinh nhận thấy vai trò quan trọng của bài toán MAP trong học máy thống kê cũng như các thách thức về việc phát triển các thuật toán hiệu quả cho bài toán. Mặc dù các nhà nghiên cứu vẫn không ngừng cải tiến, đề xuất các thuật toán đáp ứng tốt hơn cho các mô hình học máy ngày càng phức tạp nhưng vẫn còn một khoảng cách rất lớn giữa hiệu quả thực tế của các thuật toán đạt được và mong muốn của con người. Rất nhiều thuật toán đề xuất chưa đảm bảo các tiêu chuẩn như về sự hội tụ nhanh, tính phổ dụng, tính linh hoạt hay khả năng hiệu chỉnh khi áp dụng cho các mô hình thực tế phức tạp và thực hiện trên các bộ dữ liệu lớn. Do vậy, nghiên cứu các phương pháp giải hiệu quả bài toán MAP không lồi trong học máy thực sự có ý nghĩa, nhất là đặt trong bối cảnh các mô hình học máy phát triển ngày càng phức tạp với nhiều tham số hơn và thường làm việc trên các dữ liệu quan sát lớn, từ đó đòi hỏi ngày càng cao về chất lượng của các thuật toán giải.

Nhận thức được điều này, nghiên cứu sinh đặt ra bài toán cần nghiên cứu của mình là: Nghiên cứu đề xuất các thuật toán ngẫu nhiên hiệu quả giải bài toán MAP không lồi xuất hiện trong các mô hình đồ thị xác suất được cho dưới dạng:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} [f(\boldsymbol{x}) = \log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})]$$

trong đó hàm mục tiêu $f(\boldsymbol{x})$ là hàm không lồi trên miền ràng buộc Ω. Khó khăn của bài toán đặt ra ở đây chính là hàm mục tiêu $f(\boldsymbol{x})$ không lồi, có thể xuất hiện nhiều điểm cực trị địa phương/điểm yên ngựa, đồng thời $f(\boldsymbol{x})$ là hàm nhiều biến có số chiều lớn, có thể gặp khó khăn trong việc tính trực tiếp đạo hàm các cấp, do đó bài toán MAP không lồi có thể trở thành khó giải [36, 53, 54, 55].

Nghiên cứu sinh đặt ra mục tiêu là đề xuất được một số thuật toán tối ưu ngẫu nhiên để giải hiệu quả bài toán MAP không lồi đảm bảo các tiêu chí như sau:

(i) Các thuật toán ngẫu nhiên đảm bảo chất lượng về lý thuyết và thực nghiệm,

(ii) Các thuật toán có tốc độ hội tụ nhanh,

(iii) Các thuật toán có tính linh hoạt, tính tổng quát và khả năng hiệu chỉnh tốt. Từ đó có thể áp dụng các thuật toán đó rộng rãi trong nhiều mô hình trong học máy.

Để triển khai được các mục tiêu đặt ra, nghiên cứu sinh đã lựa chọn đề tài "*Một số phương pháp ngẫu nhiên cho bài toán cực đại hóa xác suất hậu nghiệm không lồi trong học máy*" cho luận án của mình. Sự thành công của đề tài góp phần giải quyết tốt hơn bài toán ước lượng MAP không lồi, đồng thời có thể mở rộng áp dụng để giải tốt các bài toán tối ưu không lồi thường xuất hiện trong nhiều mô hình học máy.

3. Các đóng góp chính của luận án

Với mục tiêu triển khai thành công đề tài, các nghiên cứu của luận án tập trung chính vào các đề xuất sau đây:

- Đề xuất bốn thuật toán tối ưu ngẫu nhiên OPE1, OPE2, OPE3 và OPE4 giải bài toán suy diễn hậu nghiệm trong mô hình chủ đề có bản chất là bài toán tối ưu không lồi thông qua việc sử dụng phân phối xác suất đều kết hợp với dùng hai chuỗi biên ngẫu nhiên xấp xỉ cho hàm mục tiêu ban đầu, trong đó các đề xuất có đảm bảo về cơ sở lý thuyết và thực nghiệm.

- Đề xuất thuật toán tối ưu ngẫu nhiên GOPE giải bài toán MAP không lồi trong mô hình chủ đề thông qua sử dụng phân phối Bernoulli với tham số $p \in (0, 1)$ thích hợp. Từ đó, áp dụng GOPE để thiết kế thuật toán ngẫu nhiên Online-GOPE học mô hình chủ đề hiệu quả.

- Sử dụng ngẫu nhiên Bernoulli với tham số $p \in (0, 1)$ thích hợp, kết hợp với dùng hai biên ngẫu nhiên và nguyên lý tham lam, nghiên cứu sinh đề xuất thuật toán ngẫu nhiên BOPE giải bài toán MAP không lồi tổng quát. BOPE được thiết kế đảm bảo các tiêu chí quan trọng của một thuật toán tối ưu mong muốn như đảm bảo tốc độ hội tụ nhanh, có tính linh hoạt dễ dàng mở rộng được cho các mô hình khác, có tính hiệu chỉnh giúp mô hình tránh được hiện tượng quá khớp. Chúng tôi đã áp dụng thành công thuật toán BOPE vào mô hình chủ đề LDA, mô hình thông dụng để giải quyết bài toán phân tích văn bản và mô hình CTMP trong hệ gợi ý.

Các thuật toán đề xuất trong luận án có ưu điểm vượt trội so với các thuật toán đã có khi xét trên một số tiêu chí quan trọng như: Thuật toán có đảm bảo cơ

sở lý thuyết cho sự hội tụ hay không? Tốc độ hội tụ là bao nhiêu? Thuộc nhóm thuật toán ngẫu nhiên không? Có khả năng linh hoạt dễ dàng mở rộng áp dụng cho các mô hình bài toán khác hay không? Có khả năng hiệu chỉnh hay không? Chi tiết kết quả đối chiếu so sánh được tổng kết trong Bảng 3 dưới đây:

Phương pháp suy diễn	**Tốc độ hội tụ**	**Ngẫu nhiên**	**Linh hoạt**	**Hiệu chỉnh**
VB [39]	–	–	–	–
CVB [40]	–	–	–	–
CVB0 [42]	–	–	–	–
CGS [43]	–	Có	–	–
CCCP [44]	–	–	–	–
SMM [45]	–	–	–	–
PMD [49]	$\mathcal{O}(T^{-1/2})$	Có	–	–
HAMCMC [50]	$\mathcal{O}(T^{-1/3})$	Có	–	–
OPE [28]	$\mathcal{O}(1/T)$	Phân phối đều	Có	–
OPE1, OPE2, OPE3, OPE4	$\mathcal{O}(1/T)$	Phân phối đều	Có	–
GOPE, BOPE	$\mathcal{O}(1/T)$	Phân phối Bernoulli	Có	Có

Bảng 3: So sánh lý thuyết của các phương pháp suy diễn trên các tiêu chuẩn như tốc độ hội tụ, tính ngẫu nhiên, tính linh hoạt, hiệu chỉnh. T biểu thị số lần lặp và '-' biểu thị "không xác định". Chúng tôi phát hiện ra rằng BOPE chiếm ưu thế nổi trội khi so sánh với các phương pháp suy diễn khác.

4. Bố cục của luận án

Với các thuật toán đề xuất đã nêu ở mục trên, luận án được kết cấu thành 4 chương với bố cục như sau:

- Chương 1 trình bày về một số kiến thức cơ sở liên quan đến luận án như bài toán MAP không lồi, tối ưu ngẫu nhiên, mô hình xác suất đồ thị, các phương pháp suy diễn trong mô hình xác suất đồ thị, mô hình chủ đề, thuật toán tối ưu ngẫu nhiên OPE. Đây là những kiến thức nền tảng cho việc phát triển các đề xuất của nghiên cứu sinh xuyên suốt trong luận án.

- Chương 2 trình bày một số đề xuất phương pháp tối ưu ngẫu nhiên cho bài toán suy diễn hậu nghiệm trong mô hình chủ đề với hàm mục tiêu không lồi. Chúng tôi đã sử dụng chiến lược ngẫu nhiên hóa hàm mục tiêu bằng phân phối xác suất đều kết hợp với hai biên ngẫu nhiên, đưa ra bốn thuật toán ngẫu nhiên mới đặt tên là OPE1, OPE2, OPE3 và OPE4. Các đề xuất mới, đặc biệt là OPE3 và OPE4, đảm bảo hiệu quả về tốc độ hội tụ và tính tương thích cao so với các tiếp cận trước đó. Tính hiệu quả này được chứng minh về mặt lý thuyết và thực nghiệm.

- Chương 3 trình bày thuật toán cải tiến mới GOPE giải bài toán MAP không lồi trong mô hình chủ đề thông qua khai thác phân phối Bernoulli với xác

suất $p \in (0, 1)$ phù hợp. Thuật toán GOPE đảm bảo tốc độ hội tụ $\mathcal{O}(1/T)$ với T là số bước lặp của thuật toán. Hơn nữa, tham số Bernoulli p góp phần làm thuật toán GOPE có tính linh hoạt thích nghi tốt trên nhiều loại dữ liệu. Sự hiệu quả của GOPE được chứng minh đầy đủ trên hai phương diện lý thuyết và thực nghiệm với hai bộ dữ liệu văn bản lớn.

- Chương 4 trình bày thuật toán cải tiến mới BOPE. Sử dụng ngẫu nhiên hóa Bernoulli kết hợp với chiến lược hai biên ngẫu nhiên đề xuất thuật toán ngẫu nhiên BOPE giải bài toán MAP không lồi tổng quát. Sự hiệu quả của BOPE được làm rõ trên nhiều phương diện lý thuyết và thực nghiệm. Ưu điểm của BOPE cũng được chỉ rõ trên các tiêu chí như sự hội tụ, tốc độ hội tụ, tính linh hoạt, tính hiệu chỉnh. Đồng thời nghiên cứu sinh đã áp dụng thành công BOPE vào mô hình LDA hay được sử dụng trong phân tích văn bản và mô hình CTMP sử dụng trong bài toán hệ gợi ý.

Với kết cấu 4 chương, luận án đã trình bày trọn vẹn các thuật toán đề xuất để giải bài toán MAP không lồi trong học máy. Như vậy, các nội dung trong luận án đã đáp ứng được các mục tiêu đề ra.

Chương 1
MỘT SỐ KIẾN THỨC NỀN TẢNG

Chương này trình bày về một số kiến thức cơ sở liên quan của luận án bao gồm: tổng quan về bài toán cực đại hóa xác suất hậu nghiệm, mô hình đồ thị xác suất và các phương pháp suy diễn, tối ưu ngẫu nhiên, mô hình chủ đề và một số thuật toán học trong mô hình chủ đề.

1.1. Tối ưu không lồi

1.1.1. Bài toán tối ưu tổng quát

Mô hình học máy thường được mô tả bởi bộ các tham số và bước học chính là đi tìm tham số tối ưu cho mô hình, từ đó dẫn về một bài toán tối ưu tham số. Nhiệm vụ của một thuật toán tối ưu trong học máy chính là tìm giá trị "tốt nhất" cho tham số của mô hình. Giả sử tập hợp các tham số mô hình được ký hiệu bằng $\mathbf{x}$, hàm đánh giá của mô hình thường được ký hiệu là $f(\mathbf{x})$. Bài toán tìm tham số "tốt nhất" được đưa về bài toán tối ưu có dạng $\min_{\mathbf{x}} f(\mathbf{x})$ hoặc $\max_{\mathbf{x}} f(\mathbf{x})$. Như vậy, *học một mô hình học máy* chính là *giải một bài toán tối ưu toán*. Do đó, tối ưu toán học, đặc biệt là tối ưu không lồi đã trở thành trung tâm của học máy [36].

Định nghĩa 1.1 (Tập lồi)**.** *Một tập $\Omega \subseteq \mathbb{R}^p$ được gọi là một tập lồi nếu*

$$\forall \boldsymbol{x}, \boldsymbol{y} \in \Omega \text{ và } 0 \leq \alpha \leq 1 \Rightarrow \alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y} \in \Omega.$$

Định nghĩa 1.2 (Hàm lồi)**.** *Một hàm số f xác định trên tập lồi Ω được gọi là hàm lồi trên Ω nếu*

$$f(\alpha\boldsymbol{x} + (1-\alpha)\boldsymbol{y}) \leq \alpha f(\boldsymbol{x}) + (1-\alpha) f(\boldsymbol{y}) \ \forall \boldsymbol{x}, \boldsymbol{y} \in \Omega \text{ và } 0 < \alpha < 1.$$

Chú ý rằng:

(i) Một hàm số f xác định trên tập lồi Ω được gọi là lõm nếu $-f$ là lồi trên Ω.

(ii) Cho f và g là các hàm lồi trên tập lồi C và D tương ứng. Khi đó các hàm số $\alpha f + \beta g$ $(\forall \alpha, \beta \geq 0)$ và $\max\{f, g\}$ cũng lồi trên $C \cap D$.

Xét bài toán tối ưu tổng quát

$$\min_{\mathbf{x} \in \Omega} f(\mathbf{x}) \tag{1.1}$$

trong đó hàm mục tiêu $f(\mathbf{x})$ là hàm trơn và không lồi trên miền đóng $\Omega \subset \mathbb{R}^p$. Khi $\Omega = \mathbb{R}^p$ thì bài toán (1.1) đưa về bài toán tối ưu không ràng buộc có dạng

$$\min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x}) \tag{1.2}$$

Do $\max_{\mathbf{x}\in\Omega} f(\mathbf{x}) = \min_{\mathbf{x}\in\Omega}[-f(\mathbf{x})]$, nên bài toán cực đại hóa

$$\max_{\mathbf{x}\in\Omega} f(\mathbf{x}) \tag{1.3}$$

được xem xét tương tự như bài toán cực tiểu hóa (1.1).

Định lý 1.1 (Điều kiện tối ưu bậc nhất)**.** *Cho hàm f xác định và khả vi trên $\mathbb{R}^p$. Nếu $\boldsymbol{x}^* \in \mathbb{R}^p$ là nghiệm cực tiểu địa phương của bài toán* (1.2) *thì $\nabla f(\boldsymbol{x}^*) = 0$.*

Định lý 1.2 (Điều kiện tối ưu bậc hai)**.** *Giả sử hàm số f khả vi liên tục hai lần trên $\mathbb{R}^p$. Khi đó:*

- *Nếu $\boldsymbol{x}^* \in \mathbb{R}^p$ là điểm cực tiểu địa phương của hàm f trên $\mathbb{R}^p$ thì $\nabla f(\boldsymbol{x}^*) = 0$ và $\nabla^2 f(\boldsymbol{x}^*) = 0$ nửa xác định dương.*
- *Ngược lại, nếu $\nabla f(\boldsymbol{x}^*) = 0$ và $\nabla^2 f(\boldsymbol{x}^*) = 0$ xác định dương thì $\boldsymbol{x}^*$ là điểm cực tiểu địa phương chặt của f trên $\mathbb{R}^p$.*

Đối với bài toán tối ưu lồi, nghiệm tối ưu địa phương cũng là tối ưu toàn cục. Do đó, tối ưu lồi đã được nghiên cứu rất đầy đủ trên khía cạnh lý thuyết và ứng dụng, đồng thời có nhiều thuật toán hiệu quả được đề xuất để giải chúng. Ngược lại, giải các bài toán tối ưu không lồi thường gặp nhiều khó khăn bởi tính đa cực trị của hàm mục tiêu. Với mỗi lớp bài toán tối ưu không lồi thường có một số phương pháp giải phù hợp đi kèm. Một trong những cách tiếp cận phù hợp và hiệu quả hiện nay chính là nhóm phương pháp dựa vào thông tin đạo hàm, trong đó có các phương pháp bậc nhất chỉ dựa vào thông tin đạo hàm cấp một, ví dụ như phương pháp GD hay SGD và các phương pháp bậc hai sử dụng đạo hàm cấp hai như phương pháp Newton và các biến thể [36]. Phương pháp bậc hai thường cho kết quả tốt hơn nhưng chi phí tính toán đạo hàm cấp hai thường tốn kém và thậm chí không tính được. Chính vì vậy, bài toán tối ưu trong học máy thường hay sử dụng *phương pháp ngẫu nhiên bậc nhất*, đảm bảo đủ đơn giản và độ chính xác cần thiết khi áp dụng.

1.1.2. Tối ưu ngẫu nhiên

Các phương pháp tối ưu tất định kinh điển thường chỉ áp dụng tốt cho bài toán tối ưu lồi và các bộ dữ liệu huấn luyện nhỏ [9, 36]. Do đó khi đối mặt với

các bài toán tối ưu không lồi, các phương pháp tất định thường kém hiệu quả. Các phương pháp *tối ưu ngẫu nhiên* như SGD [56] ra đời đã khắc phục nhược điểm của tối ưu tất định.

Mục đích của một hệ thống học là tìm tham số tối ưu thông qua tối ưu hóa một hàm đánh giá, giả sử đi tìm giá trị cực tiểu hàm *hàm kì vọng rủi ro* $J(\mathbf{w})$ như sau:

$$J(\mathbf{w}) \triangleq \mathbb{E}_{\mathbf{z}} Q(\mathbf{z}, \mathbf{w}) \triangleq \int Q(\mathbf{z}, \mathbf{w}) dP(\mathbf{z}) \tag{1.4}$$

trong đó $\mathbf{w}$ là biến cần tìm để cực tiểu hóa hàm rủi ro $J(\mathbf{w})$, $\mathbf{z}$ là các quan sát đã biết và $Q(\mathbf{z}, \mathbf{w})$ là hàm mô tả độ rủi ro của hệ thống với quan sát $\mathbf{z}$. Thông thường hàm phân phối của dữ liệu $P(\mathbf{z})$ là không biết trước, nên chúng ta phải xấp xỉ hàm kỳ vọng rủi ro $J(\mathbf{w})$ bởi *hàm rủi ro thực nghiệm* $\hat{J}_L(\mathbf{w})$ dựa trên L quan sát $\mathbf{z}_n,\ n = 1, 2, .., L$ như sau:

$$J(\mathbf{w}) \approx \hat{J}_L(\mathbf{w}) \triangleq \frac{1}{L} \sum_{n=1}^{L} Q(\mathbf{z}_n, \mathbf{w}) \tag{1.5}$$

Để cực tiểu hóa hàm $\hat{J}_L(\mathbf{w})$ ta có thể sử dụng thuật toán GD cập nhật giá trị của $\mathbf{w}$ sau mỗi vòng lặp theo công thức:

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \rho_t \bigtriangledown_{\mathbf{w}} \hat{J}_L(\mathbf{w}_t) = \mathbf{w}_t - \rho_t \frac{1}{L} \sum_{n=1}^{L} \bigtriangledown_{\mathbf{w}} Q(\mathbf{z}_n, \mathbf{w}_t) \tag{1.6}$$

Khi tốc độ học ρ_t đủ nhỏ, thuật toán hội tụ về cực tiểu địa phương. Tuy nhiên, chúng tôi thấy rằng, mỗi lần cập nhật tham số, thuật toán GD phải duyệt qua tất cả các quan sát một lần, điều này gây tốn kém về mặt thời gian và bộ nhớ khi làm việc với dữ liệu lớn. Thuật toán tối ưu ngẫu nhiên SGD đã khắc phục được nhược điểm này [56]. Ý tưởng của SGD là thay $\nabla_{\mathbf{w}} \hat{J}_L(\mathbf{w})$ bằng một hàm ngẫu nhiên $R(\mathbf{z}_i, \mathbf{w})$ thỏa mãn

$$\mathbb{E}_{Q(\mathbf{z}_i)} R(\mathbf{z}_i, \mathbf{w}) = \nabla_{\mathbf{w}} \hat{J}_L(\mathbf{w}) \tag{1.7}$$

với $\mathbf{z}_i$ là một quan sát ngẫu nhiên lấy theo phân phối đều từ tập quan sát. Khi đó, ta có $R(\mathbf{z}_i, \mathbf{w}) = \nabla_w Q(\mathbf{z}_i, \mathbf{w})$. Việc thay thế như vậy là một xấp xỉ nhiễu tới đạo hàm đúng. Tuy nhiên, với tốc độ học ρ_t phù hợp thỏa mãn điều kiện [9, 56]:

$$\sum_{t=1}^{\infty} \rho_t \to \infty \ , \ \sum_{t=1}^{\infty} \rho_t^2 < \infty \tag{1.8}$$

thì SGD hội tụ tới một cực trị địa phương theo công thức cập nhật như sau:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \rho_t \nabla_{\mathbf{w}} Q(\mathbf{z}_i, \mathbf{w}_t) \tag{1.9}$$

trong đó $\mathbf{z}_i$ là một quan sát được chọn ngẫu nhiên trong tập quan sát.

Điều này cho phép thuật toán lặp đi lặp lại giữa việc lấy mẫu dữ liệu và điều chỉnh cấu trúc ẩn dựa trên các mẫu được lấy. Một chuỗi ρ_t thỏa mãn điều kiện trên hay được sử dụng có dạng như sau:

$$\rho_t = (t + \tau)^{-\kappa} \tag{1.10}$$

trong đó $t = 1, 2, \ldots, T$ (với T là số lượng vòng lặp cần thiết). Tham số $\tau \geq 0$ là gọi là trọng số tiêu biến (decay weight) và $\kappa \in (0.5, 1)$ là tham số quên (forgetting rate). Tham số τ và κ có thể điều chỉnh thủ công sao cho thuật toán học thu được kết quả tốt nhất.

Một lưu ý là thay vì việc lấy ngẫu nhiên một quan sát, ta cũng có thể thực hiện lấy ngẫu nhiên B quan sát, tức là lấy theo mẫu nhỏ (mini-batch) $\{z_{i1}, z_{i2}, .., z_{iB}\}$. Công thức cập nhật khi sử dụng mẫu nhỏ kích thước B như sau:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \rho_t \frac{1}{B} \sum_{b=i1}^{iB} \nabla_{\mathbf{w}} Q(z_b, \mathbf{w}_t) \tag{1.11}$$

Phương pháp này sử dụng bộ nhớ ít hơn cách tiếp cận tối ưu "batch" thông thường.

Do tối ưu không lồi rất thường gặp trong học máy, nên ngày càng có nhiều nghiên cứu về các thuật toán giải bài toán không lồi trong học máy. Thuật toán SVRG (Stochastic Variance Reduced Gradient) và Proximal SVRG (Prox-SVRG) [57] có thể áp dụng cho bài toán có hàm mục tiêu là tổng hữu hạn các hàm không lồi. Tuy nhiên, SVRG và Prox-SVRG có hạn chế là có thể không hội tụ đến nghiệm toàn cục. Thuật toán CCCP (Concave-convex procedure) [44] cũng được ứng dụng rộng rãi cho bài toán không lồi. CCCP biến đổi bài toán không lồi thành tổng của các hàm lồi và hàm lõm sau đó tuyến tính hóa hàm lõm. Tuy nhiên độ phức tạp của CCCP lớn vì phải giải một bài toán quy hoạch toàn phương trong mỗi vòng lặp. GOA (graduated optimization algorithm) [58] cũng được phổ biến cho bài toán tối ưu không lồi nhưng lại đối mặt với việc tính toán trực tiếp các đạo hàm. Tác giả Hazan và các cộng sự [59] đề xuất thuật toán GradOpt (Graduated Optimization) có khả năng hội tụ đến nghiệm tối ưu toàn cục với tham số (a, σ) thích hợp. Hazan cũng chỉ ra GradOpt nhanh hơn mini-batch SGD. Các tác giả trong [60] đề xuất SVRG-GOA và PSVRG-GOA để giải bài toán không lồi dựa trên GOA, đồng thời chỉ ra GradOpt có một số hạn chế như hội tụ chậm do việc giảm của tốc độ học, điều kiện trên hàm mục tiêu là chặt khiến cho việc ứng dụng GradOpt bị hạn chế. Ngoài ra, một số thuật

toán tối ưu đã và đang áp dụng hiệu quả trong trong học máy và học sâu như Adagrad [61], RMSProp [62], Adadelta [63], Adam [64], RSAG [65] Natasha2 [66], NEON2 [67]. Bên cạnh việc đề xuất các thuật toán mới hiệu quả cải tiến về tốc độ hội tụ, các nghiên cứu về việc thoát khỏi điểm yên ngựa trong tối ưu không lồi cũng là được quan tâm bởi Dauphin và các cộng sự [68, 69] hay Rong Ge và cộng sự [53, 70, 71]. Theo tìm hiểu của chúng tôi, để *đánh giá sự hiệu quả của một thuật toán tối ưu*, thường xem xét thuật toán đó trên rất nhiều khía cạnh:

(i) Thuật toán áp dụng thành công trên lớp bài toán nào: bài toán lồi/không lồi, có ràng buộc/không ràng buộc? Ví dụ các phương pháp tất định kinh điển như GD, subgradient hay proximal GD, accelerate proximal GD áp dụng thành công trên tối ưu lồi với mẫu nhỏ, nhưng không hiệu quả khi làm việc với dữ liệu lớn và hàm không lồi. Mặc dù nhóm ngẫu nhiên như SGD có tốc độ hội tụ chậm, đạt $\mathcal{O}(T^{-1/2})$ cho bài toán tối ưu lồi và $\mathcal{O}(T^{-1/4})$ cho bài toán không lồi sau T bước lặp, lại thích hợp cho bài toán tối ưu gắn với các mô hình có dữ liệu lớn. Hoặc khi làm việc với bài toán tối ưu có ràng buộc người ta thường hay sử dụng phương pháp Online Frank-Wolfe (OFW) và các biến thể của nó [46, 47, 72].

(ii) Tốc độ hội tụ của thuật toán đạt được là bao nhiêu? Ví dụ, tốc độ hội tụ của phương pháp GD là $\mathcal{O}(1/T)$ khi giải bài toán tối ưu lồi và là $\mathcal{O}(T^{-1/2})$ khi giải bài toán tối ưu không lồi, phương pháp OPE có tốc độ hội tụ $\mathcal{O}(1/T)$ khi giải bài toán không lồi sau T bước lặp.

(iii) Thuật toán thuộc nhóm tất định hay ngẫu nhiên, ngẫu nhiên bậc không, bậc một hay bậc hai? Ví dụ GD và Frank-Wolfe (FW) thuộc nhóm tất định bậc nhất, phương pháp Newton thuộc nhóm tất định bậc hai, SGD và Stochastically Controlled Stochastic Gradient (SCSG)[73] thuộc nhóm ngẫu nhiên bậc nhất, còn Natasha2 [66] thuộc nhóm ngẫu nhiên bậc hai.

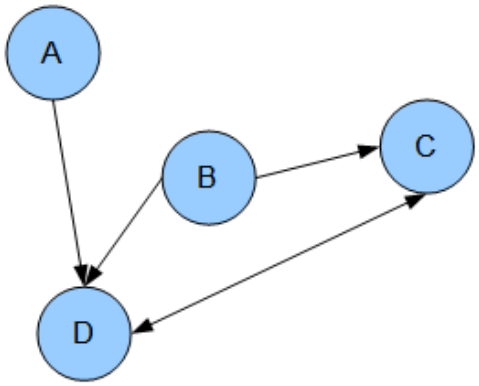
(iv) Có các giả thiết về tính trơn của hàm hay đạo hàm hay điều kiện phương sai của đạo hàm bị chặn hay không? Có nhiều phương pháp cần đến giả thiết về tính trơn của hàm mục tiêu hay gradient thỏa điều kiện Lipschitz, thậm chí cần đến giả thiết hàm trơn bậc hai. Ví dụ, GD hay SGD đều có giả thiết đạo hàm liên tục và thỏa điều kiện Lipschitz, Natasha2 hay NEON2 [67] cần đến điều kiện hàm trơn bậc hai.

(v) Chất lượng nghiệm tìm được của thuật toán đó đã đạt đến mức độ nào? Kết quả tìm được điểm dừng hay nghiệm tối ưu địa phương/toàn cục của hàm mục tiêu? Theo hiểu biết của chúng tôi, thuật toán GD, SGD, SVRG chỉ có khả năng tìm đến nghiệm xấp xỉ của điểm dừng, trong khi Natasha2 và NEON2 có thể tìm đến nghiệm xấp xỉ tối ưu địa phương đối với tối ưu không lồi [66, 67].

1.2. Mô hình đồ thị xác suất

1.2.1. Giới thiệu

Mô hình đồ thị xác suất [16, 74, 75] sử dụng đồ thị để biểu diễn phụ thuộc có điều kiện giữa các biến ngẫu nhiên một cách trực quan, trong đó có các đỉnh là các biến ngẫu nhiên, các cạnh biểu diễn sự phụ thuộc lẫn nhau của các biến ngẫu nhiên, cả đồ thị biểu diễn một phân phối đồng thời của tất cả các biến ngẫu nhiên đó. Mô hình đồ thị xác suất là một công cụ mạnh mẽ có nhiều ứng dụng trong học máy, thị giác máy tính, xử lý ngôn ngữ tự nhiên và sinh học tính toán.



Hình 1.1: Một ví dụ về một mô hình đồ thị xác suất. Mũi tên biểu trưng cho sự phụ thuộc xác suất: D phụ thuộc lần lượt vào A, B và C trong khi C phụ thuộc vào B và D.

Trong rất nhiều mô hình thống kê, khi số lượng các biến ngẫu nhiên lớn và có nhiều sự phụ thuộc của các biến, đồ thị xác suất là công cụ hữu hiệu để biểu diễn toàn bộ mô hình. Mô hình đồ thị xác suất được sử dụng rất phổ biến trong học máy thống kê do có nhiều ưu điểm:

- Mô hình ngẫu nhiên có thể được biểu diễn một cách trực quan bằng hình ảnh, giúp dễ tư duy và sử dụng;
- Việc nghiên cứu tính chất của mô hình có thể thực hiện trên đồ thị, qua

đó nhiều tính toán suy diễn có thể thực hiện hiệu quả hơn nhờ các công cụ toán học của lý thuyết đồ thị.

Phân loại có hai nhóm mô hình đồ thị xác suất chính là mạng Bayes biểu diễn quan hệ tương quan có chiều (nhân quả) thông qua một đồ thị có hướng (gọi là mô hình đồ thị có hướng) và trường Markov ngẫu nhiên chỉ biểu diễn quan hệ tương quan mà không nêu rõ quan hệ nhân quả (gọi là mô hình đồ thị vô hướng).

Nghiên cứu về mô hình đồ thị được xem xét chủ yếu theo ba phần chính: biểu diễn (cách xác định mô hình), học và suy diễn (học là làm thế nào để khớp mô hình với dữ liệu trong thế giới thực) có liên kết chặt chẽ với nhau. Để các thuật toán học và suy diễn hiệu quả, mô hình sẽ cần phải được biểu diễn đầy đủ. Hơn nữa, các mô hình học yêu cầu bước suy diễn như một chương trình/thủ tục con. Có hai cách tiếp cận suy diễn trong mô hình đồ thị xác suất, đó là suy diễn theo xác suất biên và suy diễn theo xác suất hậu nghiệm MAP. Một số nghiên cứu chỉ ra rằng suy diễn trong các mô hình đồ thị xác suất thường là khó [14, 17, 31, 38, 54] và suy diễn có khả thi hay không phụ thuộc rất nhiều vào cấu trúc đồ thị xác suất đó. Mặc dù suy diễn MAP không phải là bài toán dễ giải trong trường hợp tổng quát nhưng thấy rằng suy diễn MAP dễ giải hơn suy diễn tổng quát theo nghĩa suy diễn MAP có thể được giải trong thời gian đa thức trong khi đó suy diễn tổng quát thường thuộc loại bài toán NP-khó. Phương pháp Variable Elimination (VE) [76, 77, 78] thuộc nhóm phương pháp suy diễn chính xác, đơn giản và tổng quát trong các mô hình đồ thị xác suất, chẳng hạn như mạng Bayes và trường ngẫu nhiên Markov. Thuật toán VE có thể được sử dụng để suy diễn cực đại hóa phân phối xác suất hậu nghiệm của các biến. Tuy nhiên, thuật toán VE có độ phức tạp tính toán hàm mũ. Trong trường hợp suy diễn chính xác không khả thi, chúng ta vẫn có thể nhận được lời giải thông qua các phương pháp suy diễn xấp xỉ [14, 16, 17, 79]. Có ba phương pháp xấp xỉ điển hình để suy diễn biến ẩn từ một phân phối trong các mô hình đồ thị xác suất, đó là phương pháp suy diễn biến phân (Variational Inference) [79, 80, 81], phương pháp lan truyền kì vọng (Expectation Propagation) [80, 82] và phương pháp Monte Carlo [83].

1.2.2. Một số phương pháp suy diễn

a. Phương pháp suy diễn biến phân

Giả sử một mô hình mạng Bayes có tập các biến ẩn kí hiệu là $\mathbf{Z}$, các biến quan sát kí hiệu là $\mathbf{X}$. Mục tiêu là đi tìm một xấp xỉ cho phân phối hậu nghiệm $P(\mathbf{Z}|\mathbf{X})$. Ta có biểu diễn logarit của hàm phân phối biên tại $\mathbf{X}$ như sau:

$$\begin{aligned}\log P(\mathbf{X}) &= \int Q(\mathbf{Z}) \log(\frac{P(\mathbf{X},\mathbf{Z})}{Q(\mathbf{Z})}) d\mathbf{Z} - \int Q(\mathbf{Z}) \log(\frac{P(\mathbf{Z}|\mathbf{X})}{Q(\mathbf{Z})}) d\mathbf{Z} \\ &= \mathcal{L}(Q) + KL(Q||P)\end{aligned} \tag{1.12}$$

Chúng ta giả định các biến ẩn $\mathbf{Z}$ là biến ngẫu nhiên liên tục nên ta dùng dấu tích phân, nếu có biến nào là rời rạc thì ra chỉ việc thay tích phân bằng tổng các giá trị rời rạc của biến đó. Đại lượng $KL(Q||P)$ là một độ đo khoảng cách giữa hai phân phối Q và P (Kullback–Leibler divergence) thường được sử dụng trong lí thuyết thống kê. Một tính chất của độ đo KL là tính chất không âm: $KL(Q||P) \geq 0 \rightarrow \log P(\mathbf{X}) \geq \mathcal{L}(Q)$. Chính vì vậy đại lượng $\mathcal{L}(Q)$ được gọi là hàm cận dưới của hàm logarit phân phối trên dữ liệu $\log P(\mathbf{X})$ (gọi tắt là "log complete-data"). Chúng ta có thể làm cực đại hàm cận dưới bằng cách tối ưu theo phân phối Q. Dễ dàng thấy được cực đại của hàm cận dưới chính là hàm log complete-data khi đại lượng KL bằng 0 tức là $Q(\mathbf{Z}) = P(\mathbf{Z}|\mathbf{X})$. Tuy nhiên như đã biết thì phân phối $P(\mathbf{Z}|\mathbf{X})$ không thể tính toán được, do đó ta cần phải giới hạn miền không gian của phân phối $Q(\mathbf{Z})$ thay vì xét trên toàn bộ miền không gian của nó. Khi đó chúng ta tìm phân phối Q trên miền không gian mới gần với P nhất (coi như một xấp xỉ của P) và Q có thể tính toán được. Như vậy, bài toán đặt ra là xấp xỉ phân phối hậu nghiệm $P(\mathbf{Z}|\mathbf{X})$ bằng phân phối $Q(\mathbf{Z})$ bằng cách cực đại hàm cận dưới theo Q và có điều kiện ràng buộc của Q.

Suy diễn biến phân chính là tìm cách ràng buộc phân phối Q bằng cách phân rã phân phối này thành tích của nhiều phân phối nhỏ hơn. Giả sử các biến ẩn $\mathbf{Z}$ có thể được chia thành M nhóm không giao nhau biểu thị bởi $\mathbf{Z}_i$ $(i = 1, .., M)$. Khi đó:

$$Q(\mathbf{Z}) = \prod_{i=1}^{M} Q(\mathbf{Z_i}) \tag{1.13}$$

Phương pháp biến phân này xuất phát từ một khung xấp xỉ trong vật lý được gọi là "mean field theory" [84].

b. Phương pháp Markov Chain Monte Carlo (MCMC)

Với cách tiếp cận ước lượng các biến ẩn bằng cách xấp xỉ hàm phân phối hậu nghiệm, phương pháp suy diễn biến phân đi tìm một phân phối xấp xỉ với phân phối đó nhưng dễ tính toán suy diễn ra các biến ẩn. Ngược lại, phương pháp lấy mẫu tính kì vọng của các biến ẩn xấp xỉ dựa trên các mẫu được lấy ra từ hàm phân phối xác suất hậu nghiệm.

Lấy mẫu cơ bản: Giả sử ta có M mẫu $\{x_1, x_2, .., x_M\}$ được lấy mẫu độc lập từ phân phối P. Khi đó kì vọng của một hàm $f(x)$ theo phân phối P sẽ có thể được tính xấp xỉ bằng:

$$E_P(f) \approx \frac{1}{M} \sum_{i=1}^{M} f(x_i) \tag{1.14}$$

Một vấn đề đặt ra là làm sao có thể lấy mẫu từ phân phối P. Ta xét một ví dụ đơn giản đó là một phân phối với biến ngẫu nhiên x rời rạc nhận K giá trị trong $\{1, 2, .., K\}$. Ta có $P(x = i) = \theta_i$ với $i = 1, .., K$ và $\sum_{i=1}^{K} \theta_i = 1$. Ta thực hiện phép lấy mẫu như sau:

- Lấy một số y ngẫu nhiên phân phối đều trong khoảng $(0, 1)$,
- Chia khoảng $(0, 1)$ thành K khoảng $s_1, s_2, .., s_K$ có độ dài tương ứng là $\theta_1, \theta_2, .., \theta_K$. Kiểm tra xem y thuộc đoạn nào. Giả sử y thuộc đoạn s_j,
- Mẫu lấy được là $x = j$.

Đối với các phân phối đơn giản như Gauss, Gamma hay Dirichlet việc lấy mẫu thường đơn giản theo một công thức đã có, còn trong trường hợp phân phối P phức tạp thì phương pháp thông thường không thực hiện được. Phương pháp lấy mẫu MCMC được cho là hiệu quả trong trường hợp biến ngẫu nhiên của P có số chiều lớn và phân phối P có dạng phức tạp ví dụ như phương pháp Gibbs Sampling [85].

c. Phương pháp Gibbs Sampling

Phương pháp Gibbs Sampling (GS) [85] là một trường hợp đặc biệt của thuật toán lấy mẫu Metropolis-Hastings [86]. Giả sử ta có phân phối $P(\mathbf{z}) = P(z_1, z_2, .., z_M)$ mà ta mong muốn lấy mẫu. Phân phối này khó tính toán, tuy nhiên, xét các phân phối của một biến khi biết các biến còn lại thì lại có thể tính toán được vì chỉ có một chiều, tức là xét các phân phối $P(z_i|\mathbf{z}_{-i})$ với $i = 1, .., M$ và $\mathbf{z}_{-i} = (z_1, .., z_{i-1}, z_{i+1}, .., z_M)$. Phương pháp Gibbs Sampling được trình bày trong Thuật toán 1.1.

Thuật toán 1.1 Phương pháp Gibbs Sampling tổng quát

Đầu vào: Các phân phối $Q(z_{dn}|\mathbf{z}_{-dn})$
Đầu ra: Các mẫu được lấy từ phân phối $Q(\mathbf{z}_d)$

Khởi tạo $z_{dn}^{(0)}$ với $n = 1, 2, .., N$
for $t = 0, 1, .., \infty$ **do**
  Lấy mẫu $z_{d1}^{(t+1)} \sim Q(z_{d1}|z_{d2}^{(t)}, z_{d3}^{(t)}, .., z_{dN}^{(t)})$
  Lấy mẫu $z_{d2}^{(t+1)} \sim Q(z_{d2}|z_{d1}^{(t+1)}, z_{d3}^{(t)}, .., z_{dN}^{(t)})$
  ...
  Lấy mẫu $z_{di}^{(t+1)} \sim Q(z_{di}|z_{d1}^{(t+1)}, .., z_{di-1}^{(t+1)}, z_{di+1}^{(t)}, .., z_{dN}^{(t)})$
  ...
  Lấy mẫu $z_{dN}^{(t+1)} \sim Q(z_{dN}|z_{d1}^{(t+1)}, z_{d2}^{(t+1)}, .., z_{dN-1}^{(t+1)})$
end for

Thuật toán này có nguồn gốc từ MCMC. Bộ giá trị khởi tạo $\mathbf{z}^{(0)}$ chính là biểu diễn cho trạng thái xuất phát trên chuỗi Markov (mỗi bộ giá trị biểu diễn cho một trạng thái trên chuỗi Markov). Mỗi lần lấy mẫu z_i mới là một bước đi ngẫu nhiên thực hiện chuyển trạng thái trên chuỗi Markov. Khi đó phân bố xác suất về khả năng đi đến các trạng thái sẽ thay đổi. Cứ thực hiện bước đi ngẫu nhiên như thế, phân phối này sẽ tiến dần tiến đến một phân phối ổn định chính là phân phối $P(\mathbf{z})$, tức là các mẫu càng ở các vòng lặp sau của phương pháp GS sẽ càng gần với các mẫu đúng được lấy mẫu từ $P(\mathbf{z})$. Chính vì vậy, phương pháp GS thường bỏ qua các mẫu ở một số vòng lặp đầu tiên.

1.3. Bài toán cực đại hóa xác suất hậu nghiệm

1.3.1. Giới thiệu bài toán MAP

Chúng tôi quan tâm tới bài toán cực đại hóa ước lượng xác suất hậu nghiệm MAP không lồi trong các mô hình đồ thị xác suất. Ước lượng MAP có vai trò quan trọng trong nhiều mô hình thống kê với các biến ẩn hay các tham số không chắc chắn. Bản chất, bài toán MAP có dạng

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} \underbrace{P(\boldsymbol{x}|D)}_{\text{Posterior}} \tag{1.15}$$

trong đó biểu thức $P(\boldsymbol{x}|D)$ được gọi là *xác suất hậu nghiệm* của $\boldsymbol{x}$. Ý tưởng của MAP là chọn tham số $\boldsymbol{x}$ sao chúng gần với tham số thực của dữ liệu nhất có thể [87]. Ước lượng MAP cung cấp một cách kết hợp thông tin tiên nghiệm trong quá trình huấn luyện, đặc biệt hữu ích để xử lý các bài toán đặt ra với dữ liệu huấn luyện ít và thưa. Sự khác biệt giữa ước lượng MAP và MLE ở giả định về phân phối tiên nghiệm thích hợp của các tham số $\boldsymbol{x}$ cần ước lượng [10]. Chúng tôi nhận thấy các giả thiết hợp lý và tri thức tiên nghiệm có thể giúp cải thiện

tính chính xác của quá trình suy diễn. Thông thường chúng ta khó xác định một cách trực tiếp hàm tối ưu trong (1.15). Vì vậy, chúng ta có thể sử dụng quy tắc Bayes để đưa bài toán MAP về dưới dạng:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}}[P(D|\boldsymbol{x}) \times P(\boldsymbol{x})] \tag{1.16}$$

Phân phối tiên nghiệm $P(\boldsymbol{x})$ chính là những thông tin biết trước về $\boldsymbol{x}$. Điểm khác biệt lớn nhất giữa hai bài toán tối ưu MLE và MAP là việc hàm mục tiêu của MAP có thêm phân phối tiên nghiệm $P(\boldsymbol{x})$. Do đó, nếu chọn được phân phối tiên nghiệm phù hợp thì việc tối ưu bài toán MAP trở nên dễ giải hơn.

Trong nhiều trường hợp, vế phải của (1.16) là các xác suất nhỏ, dẫn đến hiện tượng khuếch đại sai số tính toán. Khắc phục điều này, chúng ta thường phát biểu lại bài toán MAP dưới dạng tương đương bằng cách lấy logarit của vế phải:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}}[\log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})] \tag{1.17}$$

Thông qua biểu diễn (1.17), chúng ta có thể thấy rằng MAP chính là một hiệu chỉnh của MLE với $\log P(\boldsymbol{x})$ đóng vai trò như phần hiệu chỉnh. Do đó, MAP có thể giúp mô hình tránh hiện tượng quá khớp và MAP thường mang lại hiệu quả cao hơn MLE trong trường hợp chúng ta có ít dữ liệu huấn luyện.

Bài toán MAP (1.17) có thể được xem xét dưới dạng bài toán tối ưu toán học:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}}[f(\boldsymbol{x}) = \log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})] \tag{1.18}$$

Khi đó $f(\boldsymbol{x}) = \log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})$ đóng vai trò là hàm mục tiêu của bài toán cần tối ưu. Mức độ khó giải của bài toán MAP phụ thuộc vào đặc điểm của hàm mục tiêu $f(\boldsymbol{x})$. Trong một số mô hình [27], hàm mục tiêu $f(\boldsymbol{x})$ có dạng làm lồi, nên bài toán MAP có thể được giải hiệu quả bằng các phương pháp tối ưu lồi ngay cả ở trong trường hợp số chiều lớn [8, 88]. Một khó khăn của MAP chính là hàm mục tiêu $f(\boldsymbol{x}) = \log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})$ là hàm không lồi, có thể gặp khó khăn khi tìm cực đại, dẫn đến giải trực tiếp bài toán MAP không khả thi [28, 37].

1.3.2. Một số phương pháp tiếp cận

Trong thống kê Bayes, MAP là ước lượng điểm cho một đại lượng chưa biết, chính là mốt (mode) của phân phối xác suất hậu nghiệm. MAP liên quan chặt chẽ đến MLE nhưng trong hàm mục tiêu có bổ sung thêm tri thức tiên nghiệm. Do đó, ước lượng MAP có thể xem là một hiệu chỉnh của ước lượng MLE,

hay MLE là một trường hợp đặc biệt của MAP khi không xét tới tri thức tiên nghiệm. Theo hiểu biết của chúng tôi, có một số cách tiếp cận để giải bài toán MAP như sau:

- Thông qua các phép phân tích, khi mốt của phân phối hậu nghiệm được cho dưới dạng "close-form" và đây là trường hợp prior liên hợp.
- Thông qua các phương pháp số như phương pháp gradient hoặc phương pháp Newton. Tuy nhiên, chúng thường yêu cầu các đạo hàm bậc nhất hoặc bậc hai phải tìm được bằng phương pháp giải tích hoặc bằng phương pháp số.
- Thông qua việc áp dụng thuật toán Expectation Maximization (EM).
- Thông qua các phương pháp Monte Carlo.

Để giải bài toán MAP, chúng ta có thể áp dụng các phương pháp giải đúng như phương pháp Variable Elimination (VE) [76, 77, 78]. Tuy nhiên, theo tìm hiểu của chúng tôi thì thuật toán VE có độ phức tạp tính toán hàm mũ [17]. Người ta thường áp dụng các phương pháp xấp xỉ để giải bài toán MAP. Một số phương pháp suy diễn xấp xỉ đã được đề xuất như phương pháp VB [39], CVB [40, 41], CVB0 [42], CGS [43], CCCP [44], SMM [45], FW [46], OFW [47],...Theo hiểu biết của chúng tôi, phương pháp VB, CVB hay CGS được sử dụng để ước lượng toàn bộ phân phối xác suất hậu nghiệm trong khi bài toán MAP là tìm ước lượng điểm. Ngoài ra, chúng ta có thể tiếp cận các giải bài toán MAP (1.17) theo cách nhìn của tối ưu. Khi đó, chúng ta có thể sử dụng các phương pháp tối ưu hiện đại để giải bài toán MAP. Trong một số trường hợp, bài toán MAP (1.17) có dạng là bài toán tối ưu lồi và có thể được giải tốt bằng các phương pháp tối ưu lồi [27]. Trong các mô hình đồ thị xác suất, chúng tôi thường nghiên cứu bài toán MAP trong trường hợp có số chiều lớn. Do đó, độ khó của bài toán MAP phụ thuộc vào độ phức tạp của hàm mục tiêu $f(\boldsymbol{x}) = \log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})$ rất nhiều. Nếu hàm mục tiêu $f(\boldsymbol{x})$ không lồi, việc giải bài toán tối ưu không lồi (1.18) trở nên khó khăn vì có thể gặp hiện tượng có nhiều điểm cực trị địa phương hoặc xuất hiện điểm yên ngựa [70, 65]. Các phương pháp tối ưu phổ biến như gradient descent (GD), stochastic gradient descent (SGD) hay phương pháp Newton có thể mắc kẹt trong các điểm cực trị địa phương tồi [9, 89].

Theo hiểu biết của chúng tôi, sự hiệu quả của các phương pháp suy diễn thường được đánh giá trên một số tiêu chí quan trọng như: Thuật toán có đảm

bảo cơ sở lý thuyết cho sự hội tụ hay không? Tốc độ hội tụ của thuật toán là bao nhiêu? Thuật toán thuộc nhóm ngẫu nhiên hay tất định? Thuật toán có khả năng linh hoạt (tức là có dễ dàng mở rộng áp dụng cho các mô hình khác) hay không? Thuật toán có khả năng hiệu chỉnh không? Như vậy, tìm ra các thuật toán mới hiệu quả để giải bài toán MAP (1.18) trong trường hợp $f(\boldsymbol{x})$ là một hàm không lồi là cần thiết bởi vì bài toán (1.18) nói chung là NP-khó [28, 37, 55].

Chúng tôi thấy rằng, nếu phân phối xác suất trên $\boldsymbol{x}$ và D có thể được mô tả bằng các hàm giải tích, thì bài toán MAP đưa về bài toán cực đại hóa hàm $f(\boldsymbol{x}) = g_1(\boldsymbol{x}) + g_2(\boldsymbol{x})$ trong đó $g_1(\boldsymbol{x}) = \log P(D|\boldsymbol{x})$ và $g_2(\boldsymbol{x}) = \log P(\boldsymbol{x})$. Do đó, bài toán (1.17) được đưa về bài toán tối ưu như sau

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} [f(\boldsymbol{x}) = g_1(\boldsymbol{x}) + g_2(\boldsymbol{x})] \tag{1.19}$$

Chúng ta có thể tiếp cận giải bài toán MAP (1.17) dưới dạng bài toán tối ưu không lồi (1.19). Khi đó, chúng ta có thể sử dụng các phương pháp tối ưu ngẫu nhiên hiện đại cùng với các cải tiến thích hợp để giải chúng [36]. Một minh họa cho hướng tiếp cận này chính là thuật toán OPE [28] được đề xuất để giải bài toán MAP trong các mô hình đồ thị xác suất. Chúng tôi nhận thấy OPE đảm bảo tốc độ hội tụ là $\mathcal{O}(1/T)$ vượt qua các thuật toán ngẫu nhiên đương đại để giải bài toán MAP không lồi trong các mô hình đồ thị xác suất.

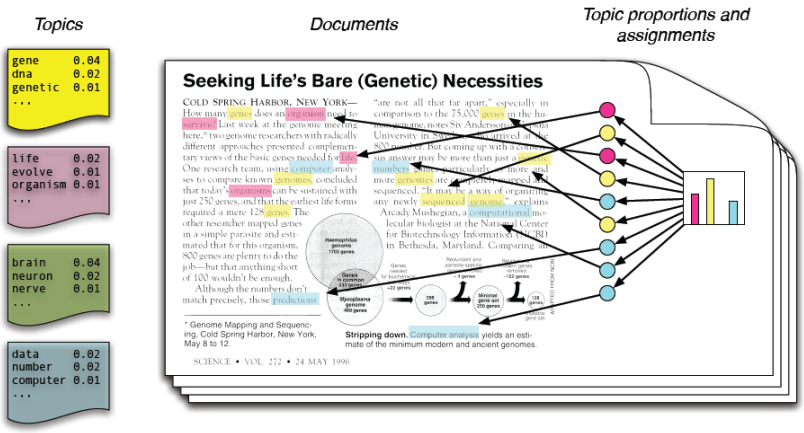
1.4. Mô hình chủ đề

1.4.1. Giới thiệu về mô hình chủ đề

Khái niệm chủ đề : Một chủ đề có thể hiểu theo nghĩa thông thường, chẳng hạn chủ đề về thể thao, văn hóa hay chủ đề về chính trị, giáo dục,..Căn cứ vào những từ xuất hiện trong văn bản mà ta có thể xác định văn bản đang nói về vấn đề gì. Ví dụ nếu trong văn bản chứa các từ: *tổng thống, chủ tịch, bầu cử, cử tri, đại biểu, nghị viện, quốc hội, tranh cử ...* thì văn bản đó được xem là một văn bản nói về *chính trị* chứ không phải là thể thao. Như vậy một chủ đề được xác định bởi một tập hợp các từ đồng thời xuất hiện để làm nổi lên chủ đề đó trong một văn bản. Để mô hình hóa bằng toán học, ta giả sử rằng mỗi từ trong tập từ điển đều xuất hiện trong một chủ đề với một xác suất nào đó. Những từ mà có xác suất trong chủ đề đó cao thì các từ đó sẽ là đặc trưng cho chủ đề đó. Trong khi đó, những từ xuất hiện với xác suất nhỏ thì ta có thể bỏ qua, coi như chúng không thuộc chủ đề đó. Nói một cách ngắn gọn hơn, mỗi chủ đề được

biểu diễn bằng một phân phối các từ trong tập từ điển. Và phân phối của các từ trên mỗi chủ đề là khác nhau để phản ánh rằng các chủ đề đó là khác nhau.

Một mô hình phân tích các chủ đề nằm trong một tập văn bản nhằm mục tiêu học ra các chủ đề ẩn này. Bằng việc xem xét cái văn bản dưới góc độ tổ hợp của các chủ đề ẩn, chúng ta có thể rút ra các đặc điểm của tập văn bản, từ đó có nhiều ứng dụng như xác định các nội dung đặc trưng nằm trong tập văn bản, phân cụm các văn bản trong tập văn bản.



Hình 1.2: Mô tả trực quan một mô hình chủ đề.

Hình 1.2 là mô tả trực quan một mô hình chủ đề. Các chủ đề (topics) được biểu diễn bởi phân phối trên các từ, những từ có xác suất xuất hiện cao nhất sẽ là đặc trưng cho chủ đề đó (4 chủ đề màu khác nhau như trong hình). Một văn bản là tổ hợp của các chủ đề ẩn với tỉ lệ đóng góp của các chủ đề khác nhau. Ví dụ văn bản trong hình vẽ có chủ đề màu vàng chiếm tỉ lệ cao nhất, tức là khả năng văn bản này nói về chủ đề này là rất cao. Hình vẽ này còn thể hiện một mức ý nghĩa đó là mức các từ. Mỗi từ trong văn bản đó được gán vào một chủ đề nào đó (các từ là các hình tròn với màu tương ứng là phép gán từ đó thuộc vào chủ đề màu đó).

Học cấu trúc ẩn của dữ liệu, mô hình phân tích ngữ nghĩa ẩn (Latent Semantic Indexing - LSI) [90] và probabilistic Latent Semantic Indexing (pLSI) [91] là lớp các phương pháp học có các văn bản và từ vựng được ánh xạ sang một không gian mới gọi là "không gian ngữ nghĩa ẩn" hay được gọi là các "chủ đề ẩn". Hai mô hình LSI và pLSI đều có số lượng tham số trong ma trận [văn bản x chủ đề] tỉ lệ với số lượng văn bản có trong tập văn bản, việc tỉ lệ tuyến tính của tham số mô hình với kích thước dữ liệu sẽ dẫn tới gia tăng kích thước lưu trữ của mô hình. Ngoài ra cả hai phương pháp đều cố định số lượng văn bản được học nên không có khả năng phân tích văn bản mới xuất hiện hoặc phải học lại tất cả từ

đầu, đồng nghĩa với việc mô hình LSI và pLSI không có tính tổng quát hóa cho dữ liệu và dễ dẫn đến hiện tượng quá khớp (overfitting). Để khắc phục những hạn chế này, mô hình chủ đề ẩn Latent Dirichlet Allocation (LDA) [39] ra đời và có ứng dụng hiệu quả vào rất nhiều bài toán phân tích dữ liệu.

1.4.2. Mô hình Latent Dirichlet Allocation

a. Các khái niệm và kí hiệu

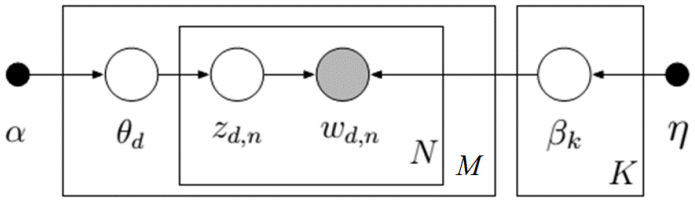
- Tập từ điển gồm V từ là đơn vị tạo thành văn bản.
- Mỗi văn bản được kí hiệu là $\mathbf{d}$. Một văn bản được biểu diễn dưới dạng vectơ đếm $\mathbf{d} = (d_1, ..., d_V)$ trong đó d_j là số lần xuất hiện của từ j trong văn bản $\mathbf{d}$.
- Mỗi văn bản là một tập hợp của các từ. Với văn bản $\mathbf{d}$, tập các từ trong văn bản đó là $\mathbf{w}_d = \{w_{d1}, w_{d2}, .., w_{dN}\}$ trong đó w_{dn} là từ thứ n trong dãy các từ của văn bản $\mathbf{d}$ với N là số lượng từ trong văn bản $\mathbf{d}$. Mỗi văn bản được biểu diễn theo túi từ (bag-of-word) chỉ quan tâm tới các từ xuất hiện mà không quan tâm tới thứ tự xuất hiện của nó trong văn bản. Tập dữ liệu $\mathcal{C}$ gồm M văn bản $\{\mathbf{d}_1, \mathbf{d}_2, .., \mathbf{d}_M\}$.
- Đơn hình $\Delta_K = \{\boldsymbol{x} \in \mathbb{R}^K : \boldsymbol{x} \geq 0, \sum_{k=1}^{K} x_k = 1\}$ và $\overline{\Delta}_K$ là phần trong của Δ_K.

b. Mô hình sinh

LDA là mô hình đồ thị xác suất trong đó quá trình tạo ra các văn bản sẽ được mô hình hóa bằng mạng Bayes: mỗi đỉnh trong mạng biểu diễn cho một biến ngẫu nhiên và mỗi cạnh nối hai đỉnh nào đó sẽ biểu diễn cho mối quan hệ xác suất giữa hai biến ngẫu nhiên đó. Các văn bản là dữ liệu có sẵn, quan sát được, chúng được sinh ra từ một quá trình tạo văn bản được biểu diễn trong mô hình. Chính vì vậy, mô hình LDA còn được gọi là mô hình sinh. Ta giả thiết rằng mỗi văn bản $\mathbf{d}$ được trộn ngẫu nhiên bởi K chủ đề ẩn với tỉ lệ các thành phần được biểu diễn bởi véc tơ tỷ lệ chủ đề $\boldsymbol{\theta}_d = (\theta_{d1}, \theta_{d2}, .., \theta_{dK})$, mỗi chủ đề ẩn k, $(k = 1, 2, .., K)$ là một phân phối xác suất trên tất cả các từ của tập từ điển. Chúng ta biểu diễn phân phối này bởi một véc tơ phân phối chủ đề $\boldsymbol{\beta}_k = (\beta_{k1}, \beta_{k2}, .., \beta_{kV})$, trong đó β_{kj} là xác suất xuất hiện của từ thứ j (theo thứ tự từ điển) thuộc vào chủ đề k. Quá trình sinh của mô hình LDA được mô tả như sau:

- Sinh ra phân phối xác suất của các từ cho mỗi chủ đề
  1. Với mỗi chủ đề k trong $\{1, \ldots, K\}$
     - Lấy mẫu $\boldsymbol{\beta}_k \sim Dir(\eta)$
- Sinh ra N từ cho một văn bản
  1. Sinh ra véc tơ tỷ lệ chủ đề của văn bản $\boldsymbol{d}$: $\boldsymbol{\theta}_d \sim \text{Dir}(\alpha)$
  2. Với $n = 1, 2, \ldots, N$ (sinh lần lượt từ thứ 1 đến từ thứ N):
     - Chọn một chủ đề $z_{dn} \sim \text{Multinomial}(\boldsymbol{\theta}_d)$ (z_{dn} là biến nhận một trong các giá trị $1, 2, \ldots, K$
     - Chọn ra từ $w_{dn} \sim \text{Multinomial}(\boldsymbol{\beta}_{z_{dn}})$

Việc giả thiết rằng véc tơ tỷ lệ chủ đề $\boldsymbol{\theta}_d$ và phân phối chủ đề $\boldsymbol{\beta}_k$ đều được sinh ra từ phân phối Dirichlet với hai tham số tương ứng là α và η với mục đích là để tăng tính tổng quát hóa của mô hình, tránh hiện tượng quá khớp (cải tiến hơn so với pLSI). Ở đây η và α được gọi là các tri thức tiên nghiệm hay gọi là prior của phân phối Dirichlet. Mô hình LDA được biểu diễn bằng đồ thị xác suất như Hình 1.3. Các kí hiệu mũi tên biểu diễn xác suất có điều kiện. Các từ $\mathbf{w}$ là đối tượng có thể quan sát được nên sẽ được tô đậm. Trong đó α và η là hai siêu tham số, các biến $\boldsymbol{\beta}$, $\boldsymbol{\theta}$, $\mathbf{z}$ và $\mathbf{w}$ là các biến ngẫu nhiên mong muốn ước lượng. Như vậy mô hình LDA gồm có ba phân mức:



Hình 1.3: Mô hình chủ đề ẩn LDA

- Mức toàn cục: siêu tham số η và α đặc trưng cho mô hình, biến $\boldsymbol{\beta}$ biểu diễn các chủ đề đặc trưng cho tập văn bản,
- Mức văn bản: biến $\boldsymbol{\theta}$ xác định cho mỗi văn bản,
- Mức từ: Các chủ đề mà mỗi từ có thể thuộc vào $\mathbf{z}$ cùng với các từ quan sát được $\mathbf{w}$,

trong đó mức văn bản và mức từ ta có thể gọi chung là mức cục bộ. Các biến phân phối chủ đề $\boldsymbol{\beta}$, tỷ lệ chủ đề $\boldsymbol{\theta}$, biến chủ đề $\mathbf{z}$ được gọi là các biến ẩn biểu

diễn cho các cấu trúc ngữ nghĩa ẩn cần khai phá từ tập văn bản quan sát được. Như vậy, bài toán học mô hình LDA chính là đi ước lượng các biến ẩn này khi đã biết các từ của các văn bản. Từ mô hình đồ thị xác suất, công việc này chính là ước lượng một phân phối hậu nghiệm, có bản chất là phân phối có điều kiện của các biến ẩn khi đã biết các biến dữ liệu và các siêu tham số:

$$P(\boldsymbol{\theta}, \mathbf{z}, \boldsymbol{\beta}|\mathbf{w}, \alpha, \eta) = \frac{P(\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}, \boldsymbol{\beta}|\alpha, \eta)}{P(\mathbf{w}|\alpha, \eta)} \tag{1.20}$$

trong đó:

$$P(\mathbf{w}|\alpha, \eta) = \int_{\boldsymbol{\beta}} \int_{\boldsymbol{\theta}} \sum_{\mathbf{z}} P(\mathbf{w}, \boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{z}|\alpha, \eta)$$

$$P(\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}, \boldsymbol{\beta}|\alpha, \eta) = \prod_{d=1}^{M} (P(\boldsymbol{\theta}_d|\alpha) \prod_{n=1}^{N} P(z_{dn}|\boldsymbol{\theta}_d) P(w_{dn}|z_{dn}, \boldsymbol{\beta})) \prod_{k=1}^{K} P(\boldsymbol{\beta}_k|\eta)$$

Việc tính toán trực tiếp phân phối hậu nghiệm theo (1.20) là không khả thi theo [37]. Vì vậy, ta cần phải tính xấp xỉ phân phối hậu nghiệm này để có thể suy diễn ra các biến ẩn.

1.4.3. Suy diễn hậu nghiệm trong mô hình chủ đề

Theo hiểu biết của chúng tôi, vấn đề chính của các mô hình đồ thị xác suất là tính phân phối hậu nghiệm của các biến ẩn với điều kiện đã biết các biến quan sát và siêu tham số của mô hình. Với mô hình chủ đề LDA, phân phối hậu nghiệm chính là $P(\boldsymbol{\theta}, \mathbf{z}|\mathbf{w}, \alpha, \boldsymbol{\beta})$ cho mỗi văn bản $\mathbf{d}$. Bài toán tính phân phối xác suất này gọi là *bài toán suy diễn*. Bài toán suy diễn có hai vai trò quan trọng như sau:

(i) Tính được phân phối hậu nghiệm giúp biểu diễn ngữ nghĩa ẩn trong mỗi văn bản. Véc tơ $\boldsymbol{\theta}$ biểu diễn tỉ lệ các chủ đề trong văn bản, $\mathbf{z}$ biểu diễn chủ đề của các từ trong văn bản. Như vậy có thể hiểu được văn bản này nói về chủ đề gì, và các chủ đề đó được gán cho mỗi từ trong văn bản như thế nào?

(ii) Tính được phân phối hậu nghiệm là bước quan trọng trong quá trình học tham số của mô hình. Với các mô hình có tham số ẩn, việc học được tham số của mô hình thông qua thuật toán EM, trong đó bước E là bước tính phân phối hậu nghiệm của các biến ẩn, bước M là cập nhật tham số cho mô hình dựa vào các biến ẩn tính được từ bước E. Do đó, *bài toán suy diễn là bài toán cốt lõi của các mô hình chủ đề*. Tốc độ của thuật toán suy diễn ảnh hưởng rất lớn đến tốc độ hội tụ của thuật toán học.

Tuy nhiên, phân phối hậu nghiệm trong các mô hình chủ đề thường không tính được dưới dạng công thức tường minh, do đó không thể biểu diễn nó và ước lượng các biến ngẫu nhiên bằng các công cụ giải tích. Có hai cách chính để giải quyết bài toán này là:

- Xấp xỉ bằng một phân phối dễ tính hơn, hoặc lấy mẫu ngẫu nhiên từ phân phối đó để ước lượng các biến ngẫu nhiên ta muốn ước lượng. Để xấp xỉ phân phối hậu nghiệm trong mô hình LDA, các tác giả đã đề xuất các phương pháp suy diễn biến phân Variational Bayes (VB) [39], Collapsed Variational Inference (CVB và CVB0) [40, 42]. Phương pháp xấp xỉ này dựa trên bất đẳng thức Jensen để ước lượng cận dưới của phân phối đang xét, sau đó cực đại hóa hàm cận dưới này để tìm một phân phối gần nhất.
- Lấy mẫu ngẫu nhiên, thuật toán lấy các mẫu ngẫu nhiên theo phân phối đang xét, sau đó ước lượng các biến ngẫu nhiên đựa trên tập mẫu này. Điển hình cho lấy mẫu ngẫu nhiên là thuật toán Collapsed Gibbs Sampling (CGS) [92].

Trong mô hình LDA, phân phối hậu nghiệm của biến ẩn cho mỗi văn bản $\boldsymbol{d}$ là:

$$P(\boldsymbol{\theta}, \mathbf{z}|\mathbf{w}, \alpha, \boldsymbol{\beta}) = \frac{P(\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}|\alpha, \boldsymbol{\beta})}{P(\mathbf{w}|\alpha, \boldsymbol{\beta})}$$

Phân phối này là không tính được vì $P(\mathbf{w}|\alpha, \boldsymbol{\beta})$ không biểu diễn được dưới dạng tường minh [39]. Mục đích chính của việc tính phân phối hậu nghiệm là ước lượng kì vọng của biến ngẫu nhiên $\boldsymbol{\theta}$ và $\mathbf{z}$. Biến $\boldsymbol{\theta}$ biểu diễn tỉ lệ chủ đề của văn bản, biến $\mathbf{z}$ biểu diễn việc các từ trong văn bản được gán cho các chủ đề nào. Phương pháp VB [39] xấp xỉ phân phối biến phân bằng một phân phối dễ tính hơn $Q(\boldsymbol{\theta}, \boldsymbol{z}|\gamma, \boldsymbol{\phi})$ với các tham số cần tìm là γ và $\boldsymbol{\phi}$. Phương pháp lấy mẫu ngẫu nhiên CGS [92] lấy mẫu trực tiếp biến ngẫu nhiên $\mathbf{z}$ theo phân phối $P(\mathbf{z}|\mathbf{w}, \alpha, \boldsymbol{\beta})$, sau đó ước lượng tỉ lệ chủ đề $\boldsymbol{\theta}$ của văn bản dựa trên các mẫu lấy được.

a. Phương pháp Variational Bayes

Phương pháp Variational Bayes (VB) [39] là một trong những phương pháp suy diễn đầu tiên để giải bài toán suy diễn hậu nghiệm với mô hình chủ đề. Bài toán học trong LDA chính là ước lượng phân phối đồng thời $P(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\beta}|\mathcal{C})$ khi cho bởi tập dữ liệu $\mathcal{C}$. Tuy nhiên, bài toán suy diễn với mô hình chủ đề là không khả thi trong một số trường hợp xấu [37]. Để khắc phục tính không khả thi, phương pháp VB giả thiết rằng các biến ẩn là độc lập. Đặc biệt sử dụng phân phối phân

Thuật toán 1.2 VB: Variational Bayes

Đầu vào: Văn bản $\mathbf{d}$ và tham số mô hình $\{\boldsymbol{\lambda}, \alpha\}$
Đầu ra: $\boldsymbol{\phi}$

```
Khởi tạo φ ngẫu nhiên
for l = 0, 1, .., ∞ do
    λ_k := α + Σ_{d_j>0} φ_jk d_j
    φ_jk ∝ exp ψ(γ_k). exp[ψ(λ_kj) − ψ(Σ_t λ_kt)]
end for
```

rã Q đơn giản hơn để ước lượng cho phân phối đồng thời $P(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\beta}|\mathcal{C})$, trong đó

$$Q(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\beta}) = \prod_{\mathbf{d}\in\mathcal{C}} Q(\mathbf{z}_d|\boldsymbol{\phi}_d) \prod_{d\in\mathcal{C}} Q(\boldsymbol{\theta}_d|\gamma_d) \prod_k Q(\boldsymbol{\beta}_k|\boldsymbol{\lambda}_k)$$

Vì vậy, bài toán học được đưa về bài toán ước lượng tham số biến phân $\{\boldsymbol{\phi}, \gamma, \boldsymbol{\lambda}\}$ bằng việc cực đại hóa hàm cận dưới trên likelihood $P(\mathcal{C}|\alpha, \eta)$, ví dụ:

$$\max \mathbf{E}_{Q(\mathbf{z},\boldsymbol{\theta},\boldsymbol{\beta})}[\log P(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\beta}, \mathcal{C}|\alpha, \eta)] + H(Q(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\beta}))$$

trong đó $H(x)$ ký hiệu cho entropy của x. Chú ý rằng VB giả thiết rằng $\boldsymbol{\beta}_k \sim Dir(\boldsymbol{\lambda}_k)$. Phương pháp VB suy diễn hậu nghiệm cho một văn bản thông qua ước lượng $P(\mathbf{z}, \boldsymbol{\theta}|\mathbf{d}, \boldsymbol{\beta}, \alpha)$. Phương pháp VB được mô tả chi tiết trong Thuật toán 1.2.

b. Phương pháp Collapsed variational Bayes

Phương pháp Collapsed variational Bayes (CVB) [40] sử dụng

$$Q(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\beta}) = Q(\boldsymbol{\theta}, \boldsymbol{\beta}|\mathbf{z}, \gamma, \boldsymbol{\lambda}) \prod_{\mathbf{d}\in\mathcal{C}} Q(\mathbf{z}_d|\boldsymbol{\phi}_d)$$

để xấp xỉ cho phân phối $P(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\beta}|\mathcal{C})$. Khi đó bài toán đưa về

$$\max E_{Q(\mathbf{z})Q(\boldsymbol{\theta},\boldsymbol{\beta}|\mathbf{z})}[\log P(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\beta}, \mathcal{C}|\alpha, \eta)] + H(Q(\mathbf{z})Q(\boldsymbol{\theta}, \boldsymbol{\beta}|\mathbf{z}))$$

Chúng ta tiến hành cực đại tối đa hóa hàm mục tiêu liên quan đến $Q(\boldsymbol{\theta}, \boldsymbol{\beta}|\mathbf{z})$ trước và tiếp theo bởi $Q(\mathbf{z})$ cho đến khi hội tụ. Phương pháp CVB có thể đưa ra xấp xỉ tốt hơn VB bởi vì duy trì sự phụ thuộc giữa $\mathbf{z}$ và $(\boldsymbol{\theta}, \boldsymbol{\beta})$. Ý tưởng từ lấy mẫu Gibbs, phương pháp CVB khai thác các token riêng rẽ trong các văn bản để suy diễn. Trong khi VB duy trì phân phối biến phân $\boldsymbol{\gamma} = (\gamma_1, ., \gamma_K)$ cho mỗi văn bản, CVB duy trì $\boldsymbol{\gamma}$ cho mỗi token. CVB làm việc tốt hơn VB. Khi dùng phương pháp CVB suy diễn cho một văn bản $\boldsymbol{d}$ cụ thể, chúng tôi thấy rằng CVB trên thực tế ước lượng $P(\mathbf{z}|\mathbf{d}, \alpha, \eta)$ đơn giản hơn $P(\mathbf{z}, \boldsymbol{\theta}|\mathbf{d}, \alpha, \eta)$ trong VB. Tuy nhiên, suy diễn hậu nghiệm bằng CVB không phải là cục bộ cho một tài liệu cụ thể và yêu cầu một số cập nhật cho các biến toàn cục. Suy diễn hậu nghiệm bởi

Thuật toán 1.3 CVB: Collapsed Variational Bayes

Đầu vào: Văn bản **d** và tham số mô hình $\{N, \alpha, \eta\}$
Đầu ra: $\boldsymbol{\phi}$

Khởi tạo ϕ ngẫu nhiên.
for $l = 0, 1, .., \infty$ **do**
  for Với token thứ i, z_i trong văn bản ***d*** **do**
    $\lambda_k^{-i} := \alpha + \sum_{t \neq i} \phi_{tk}$
    $V_k^{-i} := \sum_{t \neq i} \phi_{tk}(1 - \phi_{tk})$
    $N_{kz_i}^{-i} := N_{kz_i}^{-i} + \phi_{ik}$
    $\alpha_k^{-i} := \sum_t N_{kz_i}^{-i}$
    $X := -\frac{V_k^{-i}}{2(\gamma_k^{-i})^2} - \frac{V_{kz_i}^{i}}{2(N_{kz_i}^{-i}+\eta)^2} + \frac{V_k^{-i}}{2(\alpha_k^{-i}+V\eta)^2}$
    $\phi_{jk} \propto \gamma_k^{-i}(N_{kz_i}^{-i} + \eta)(\alpha_k^{-i} + V\eta)^{-1} \exp X$
  end for
end for

Thuật toán 1.4 CVB0: Một biến thể nhanh của CVB

Đầu vào: Văn bản **d** và tham số mô hình $\{N, \alpha, \eta\}$
Đầu ra: $\boldsymbol{\phi}$

Khởi tạo $\boldsymbol{\phi}$ ngẫu nhiên
for $l = 0, 1, ...\infty$ **do**
  for Với token thứ i, z_i trong văn bản ***d*** **do**
    $\lambda_k^{-i} := \alpha + \sum_{t \neq i} \phi_{tk}$
    $N_{kz_i}^{-i} := N_{kz_i}^{-i} + \phi_{ik}$
    $\alpha_k^{-i} := \sum_t N_{kz_i}^{-i}$
    $\phi_{jk} \propto \gamma_k^{-i}(N_{kz_i}^{-i} + \eta)(\alpha_k^{-i} + V\eta)^{-1}$
  end for
end for

CVB được trình bày chi tiết trong Thuật toán 1.3, trong đó N_{kj} đóng vai trò tương tự với λ_{kj} trong phương pháp VB.

c. Fast collapsed variational Bayes

Phương pháp fast collapsed variational Bayes (CVB0) [42] là phiên bản cải tiến của CVB. Bản cập nhật cho ϕ_{ik} trong CVB sử dụng khai triển Taylor bậc hai. Tác giả Asuncion và cộng sự [42] đã đề xuất chỉ sử dụng thông tin bậc không cho việc xấp xỉ để thực hiện việc cập nhật ϕ_{ik} đơn giản hơn đáng kể. Thuật toán CVB0 suy diễn hậu nghiệm cho một văn bản được trình bày chi tiết trong Thuật toán 1.4.

d. Phương pháp Collapsed Gibbs sampling

Ban đầu, collapsed Gibbs sampling (CGS) được đề xuất bởi [92] cho việc học LDA từ dữ liệu. Gần đây, CGS được điều chỉnh thành công cho bài toán suy diễn hậu nghiệm trên một văn bản [93]. Thuật toán CGS tiến hành ước lượng $P(\mathbf{z}|\mathbf{d}, \alpha, \eta)$ làm cho suy diễn cục bộ hơn, tức là, suy diễn hậu nghiệm cho một

Thuật toán 1.5 CGS: collapsed Gibbs sampling

Đầu vào: Văn bản $\mathbf{d}$ và tham số mô hình $\{N, \alpha, \eta\}$
Đầu ra: $\boldsymbol{\phi}$

```
Khởi tạo ϕ ngẫu nhiên.
Loại bỏ B burn-in sweeps
for l = 1, .., S mẫu do
    for Với token thứ i z_i trong văn bản d do
        λ_k^{-i} := α + Σ_{t≠i} 𝕀(z_t = k)
        φ_jk ∝ γ_k^{-i} exp[ψ(λ_{kz_i}) − ψ(Σ_t λ_kt)]
        Lấy mẫu z_i từ phân phối Multinomial(φ_i)
    end for
end for
```

văn bản không cần phải sửa đổi bất kỳ trên biến toàn cục. Đặc điểm này tương tự với VB, nhưng rất khác với CVB và CVB0. Chi tiết thuật toán CGS được trình bày trong Thuật toán 1.5.

1.5. Thuật toán OPE

Xét bài toán suy diễn hậu nghiệm đối với từng văn bản $\boldsymbol{d}$ trong mô hình chủ đề. Để ước lượng tỷ lệ chủ đề $\boldsymbol{\theta}$ cho một văn bản, có một cách tiếp cận khác chính là cực đại hóa phân phối hậu nghiệm MAP của $\boldsymbol{\theta}$. Ước lượng $\boldsymbol{\theta}$ cho một văn bản là:

$$\boldsymbol{\theta^*} = \arg\max_{\boldsymbol{\theta}\in\Delta_K} P(\boldsymbol{\theta}|\mathbf{w}, \alpha, \boldsymbol{\beta}) \tag{1.21}$$

Tuy nhiên, đây là bài toán tối ưu không lồi khó giải [37], do đó không có thuật toán trực tiếp giải bài toán này, tức là khó tìm được cực trị toàn cục. Để giải bài toán tối ưu không lồi, các tác giả sử dụng phương pháp tối ưu ngẫu nhiên để tìm nghiệm xấp xỉ cho bài toán. Và một thuật toán tối ưu ngẫu nhiên giải hiệu quả bài toán suy diễn hậu nghiệm này chính là thuật toán OPE [28].

Để ước lượng tỉ lệ chủ đề $\boldsymbol{\theta} \in \Delta_K$ cho một văn bản $\boldsymbol{d}$, chúng ta cần giải bài toán sau:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Delta_K} P(\boldsymbol{d}, \boldsymbol{\theta}|\boldsymbol{\beta}, \alpha)$$

Theo công thức Bayes có:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Delta_K} [\log P(\boldsymbol{d}|\boldsymbol{\theta}, \boldsymbol{\beta}) + \log P(\boldsymbol{\theta}|\alpha)] \tag{1.22}$$

Cho văn bản $\boldsymbol{d}$, xác suất để từ j xuất hiện trong $\boldsymbol{d}$ có thể được biểu diễn

$$P(w = j|\boldsymbol{d}) = \sum_{k=1}^{K} P(w = j|z = k)P(z = k|\boldsymbol{d}) = \sum_{k=1}^{K} \beta_{kj}\theta_k$$

Vì vậy log likelihood của $\boldsymbol{d}$ là:

$$\log P(\boldsymbol{d}|\boldsymbol{\theta}, \boldsymbol{\beta}) = \log \prod_j P(w = j|\boldsymbol{d})^{d_j} = \sum_j d_j \log P(w = j|\boldsymbol{d}) = \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj}$$

Mật độ xác suất của phân phối Dirichlet K chiều với tham số α là $P(\boldsymbol{\theta}|\alpha) \propto \prod_{k=1}^{K} \theta_k^{\alpha-1}$. Do đó, với giả thiết về quá trình sinh của LDA, bài toán (1.22) tương ứng với bài toán sau:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta} \in \Delta_K} \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^{K} \log \theta_k \tag{1.23}$$

trong đó α là tham số của phân phối tiên nghiệm Dirichlet.

Mặt khác, mức độ khó giải hay dễ giải của bài toán (1.23) phụ thuộc vào giá trị của tham số α. Với $\alpha \geq 1$, bài toán (1.23) có hàm mục tiêu là tổng hai hàm lõm, nên bài toán (1.23) được giải tốt bằng các thuật toán tối ưu lồi với thời gian đa thức. Các tác giả trong [52] đã đề xuất chuyển bài toán (1.23) về bài toán tối ưu lồi bằng cách cố định tham số $\alpha = 1$, khi đó hàm mục tiêu $f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj}$ có dạng hàm lõm. Khi đó chúng ta bỏ đi tri thức Dirichlet của $\boldsymbol{\theta}$, thay vào đó $\boldsymbol{\theta}$ là biến ngẫu nhiên có phân phối đều và mô hình LDA có thể suy biến thành mô hình FSTM [51]. Tác giả trong [52] dùng thuật toán Frank-Wolfe [94] để tối ưu hàm lõm $f(\boldsymbol{\theta})$ trong trường hợp này. Mặc dù đặt cố định tham số $\alpha = 1$ làm thiếu tính tổng quát của mô hình có thể dẫn đến hiện tượng quá khớp. Tuy nhiên, [52] cũng chỉ ra một số tính chất thú vị của thuật toán suy diễn này, đó là tạo ra nghiệm có tính chất thưa và thuật toán hội tụ với bậc tuyến tính.

Vai trò của tham số α trong mô hình LDA có thể hiểu như sau: khi $\alpha < 1$ phần lớn phân bố xác suất ở các đỉnh của đơn hình, tạo ra hầu hết văn bản thuộc một số lượng nhỏ các chủ đề. Ngược lại, khi $\alpha > 1$ hầu hết các văn bản thuộc hầu hết các chủ đề. Vì vậy, trong thực tế, khi sử dụng mô hình LDA, người ta thường chọn $\alpha < 1$ dẫn đến hàm mục tiêu của (1.23) là không lõm [92, 42]. Đó là lý do tại sao bài toán (1.23) không khả thi trong trường hợp xấu. Xét hàm mục tiêu của (1.23) có dạng:

$$f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj} - (1 - \alpha) \sum_{k=1}^{K} \log \theta_k$$

có dạng là hiệu của hai hàm lõm. Bài toán (1.23) đưa về bài toán quy hoạch DC, một lớp bài toán tối ưu không lồi điển hình. Do đó, bài toán (1.23) thuộc lớp

bài toán không lồi khó giải [37]. Đối với bài toán không lồi có ràng buộc như bài toán (1.23) thì tìm được nghiệm tối ưu toàn cục là rất khó. Trong trường hợp tìm được nghiệm thì đa phần nghiệm đó chỉ là điểm cực trị địa phương hoặc là điểm dừng.

Chúng ta có thể sử dụng một số kỹ thuật phổ biến như phương pháp nhánh cận [95], thuật toán siêu phẳng cắt, hay gần đây là thuật toán DCA [96] để giải bài toán quy hoạch DC tổng quát. Tuy nhiên, các thuật toán này không hiệu quả khi được áp dụng cho bài toán suy diễn hậu nghiệm (1.23), bản chất một bài toán tối ưu không lồi. Lấy tư tưởng của thuật toán Online Frank-Wolfe [47], các tác giả trong [97] đã đề xuất thuật toán Online Frank-Wolfe (OFW) giải bài toán suy diễn MAP không lồi với mô hình LDA. Chi tiết của thuật toán OFW được trình bày trong Thuật toán 1.6.

Thuật toán 1.6 OFW: Online Frank-Wolfe cho bài toán suy diễn MAP

Đầu vào: Văn bản $\mathbf{d}$ và tham số mô hình $\{\boldsymbol{\beta}, \alpha\}$
Đầu ra: $\boldsymbol{\theta}$ làm cực đại hàm $f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^K \log \theta_k$

Khởi tạo $\boldsymbol{\theta_1}$ trong $\overline{\Delta}_K$
for $t = 1, 2, ..\infty$ **do**
  Lấy f_t có phân phối đều từ $\{\sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}; (\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
  $F_t := \frac{2}{t} \sum_{h=1}^t f_h$
  $i' := \arg\max_i \nabla F_t(\theta_t)_i$
  $a := \dfrac{1}{\sqrt{t}}$
  $\boldsymbol{\theta}_{t+1} := a e_{i'} + (1 - a)\theta_t$
end for

Các tác giả trong bài báo [97] đã chứng minh được tốc độ hội tụ của thuật toán OFW ít nhất là $\mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$ với T là số bước lặp của thuật toán. Các tác giả trong [28] tiếp tục đề xuất thuật toán cải tiến mới là Online maximum a Posteriori Estimation (OPE). OPE có nhiều ưu điểm so với các đề xuất trước đó. Chi tiết của OPE được trình bày trong Thuật toán 1.7.

Thuật toán 1.7 OPE: Online Maximum a Posteriori Estimation

Đầu vào: Văn bản $\mathbf{d}$ và mô hình $\{\boldsymbol{\beta}, \alpha\}$
Đầu ra: $\boldsymbol{\theta}^*$ là cực đại của hàm $f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^K \log \theta_k$

Khởi tạo $\boldsymbol{\theta}_1$ thuộc $\overline{\Delta}_K = \{\mathbf{x} \in \mathbb{R}^K : \sum_{k=1}^K x_k = 1, \mathbf{x} \geq \epsilon > 0\}$
for $t = 1, 2, .., \infty$ **do**
  Lấy f_t có phân phối đều từ $\{\sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}; (\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
  $F_t := \frac{2}{t} \sum_{h=1}^t f_h$
  $\boldsymbol{e}_t := \arg\max_{\boldsymbol{x} \in \overline{\Delta}_K} < F_t'(\boldsymbol{\theta}_t), \boldsymbol{x} >$
  $\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}$
end for

Ký hiệu

$$g_1(\boldsymbol{\theta}) := \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj} \ , \ g_2(\boldsymbol{\theta}) := (\alpha - 1) \sum_{k=1}^{K} \log \theta_k$$

Hàm mục tiêu $f(\boldsymbol{\theta}) = g_1(\boldsymbol{\theta}) + g_2(\boldsymbol{\theta})$. OPE hoạt động theo cách thức như sau: tại mỗi bước lặp t, xây dựng hàm ngẫu nhiên F_t xấp xỉ cho hàm mục tiêu $f(\boldsymbol{\theta})$, tiến hành cập nhật $\boldsymbol{\theta}$ dựa trên hàm xấp xỉ này theo công thức cập nhật của Frank-Wolfe. OPE là thuật toán ngẫu nhiên áp dụng cho hàm không lồi. Vì tính chất của hàm không lồi, điểm dừng có thể là điểm yên ngựa hoặc cực trị địa phương. OPE đưa thêm tính ngẫu nhiên vào để hi vọng thuật toán nhảy ra được khỏi cực trị địa phương và điểm yên ngựa mà không phụ thuộc vào điểm khởi tạo.

Cách xây dựng hàm F_t khác biệt so với các thuật toán khác. Tại bước lặp t, hàm $F_t = \frac{2}{t} \sum_{h=1}^{t} f_h$ với f_h có phân phối đều từ $\{\sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj}; (\alpha - 1) \sum_{k=1}^{K} \log \theta_k\}$. Như vậy, tại mỗi bước lặp, OPE không có tính chất là giá trị tại điểm $\boldsymbol{\theta}_t$ của hàm xấp xỉ và hàm mục tiêu bằng nhau, mà dãy xấp xỉ ngẫu nhiên F_t hội tụ về hàm mục tiêu khi thuật toán hội tụ. Gọi a_t và b_t lần lượt là số lần chọn được $g_1(\boldsymbol{\theta})$ và $g_2(\boldsymbol{\theta})$ sau t vòng lặp. Ta có:

$$F_t = \frac{2}{t}(a_t g_1 + b_t g_2)$$

$$F_t - f = \frac{a_t - b_t}{t}(g_1 - g_2)$$

Có thể thấy, tại mỗi bước lặp t, OPE làm việc với hàm xấp xỉ F_t là tổng của hàm mục tiêu $f(\boldsymbol{\theta})$ và nhiễu ngẫu nhiên $\frac{a_t - b_t}{t}(g_1 - g_2)$ tiến về 0. Đại lượng nhiễu này làm cho OPE hoạt động tốt hơn với các hàm mục tiêu không lồi với dữ liệu thực tế. Việc chứng minh sự hội tụ của thuật toán đã được chỉ ra trong [28]. Có thể tổng kết các ưu điểm vượt trội của thuật toán OPE như sau:

- Tốc độ hội tụ của thuật toán nhanh;
- Việc xấp xỉ hàm mục tiêu đúng bằng một dãy các hàm ngẫu nhiên, với mong muốn tăng xác suất để thuật toán đi tới một nghiệm tối ưu cục bộ hoặc ít nhất cũng thoát khỏi những điểm yên ngựa "saddle point";
- Nghiệm thu được của bài toán mang tính thưa. Đây là một ưu điểm lớn khi đối mặt với bài toán có số chiều lớn.

1.6. Một số thuật toán ngẫu nhiên học LDA

LDA [39] là mô hình chủ đề điển hình có nhiều ứng dụng, đặc biệt trong phân tích văn bản. Do đó, nghiên cứu các thuật toán học mô hình LDA là quan tâm của nhiều nhà nghiên cứu [39, 40, 98, 93]. Như chúng tôi đã trình bày, mục tiêu của chúng tôi là đi ước lượng được các biến ẩn $\boldsymbol{\beta}$, $\boldsymbol{\theta}$, $\mathbf{z}$ dựa vào hàm phân phối hậu nghiệm $P(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\beta}|\mathbf{w}, \alpha, \eta)$. Các biến ẩn này có thể được chia làm 2 mức:

- Mức biến cục bộ: sẽ là các đại lượng đại diện cho biến ẩn ở mức văn bản như $\mathbf{z}$ và $\boldsymbol{\theta}$,
- Mức biến toàn cục: là đại lượng đại diện cho biến ẩn của cả tập văn bản đó là phân phối chủ đề $\boldsymbol{\beta}$.

Hướng tiếp cận ngẫu nhiên được coi là hiệu quả đối với các mô hình học máy thường làm việc với dữ liệu lớn [9, 36, 99]. Do đó trong luận án nghiên cứu sinh quan tâm tới các phương pháp ngẫu nhiên học mô hình chủ đề LDA. Các phương pháp học ngẫu nhiên cho LDA được thiết kế theo tư tưởng của phương pháp lặp, mà mỗi vòng lặp thường gồm 2 bước:

- Bước 1: Thực hiện suy diễn biến cục bộ cho từng văn bản khi đã biết biến toàn cục.
- Bước 2: Thực hiện cập nhật lại biến toàn cục theo thuật toán tối ưu ngẫu nhiên.

Sử dụng các thuật toán suy diễn như Variational Bayes (VB) [39], Collapsed variational Bayes (CVB0) [42], Collapsed Gibbs sampling (CGS) [43], các phương pháp ngẫu nhiên như Online-CGS [43], Online-CVB0 [100], Online-VB [32] đã được đề xuất để học mô hình LDA. Các tác giả trong [28] sử dụng OPE làm cốt lõi suy diễn và lược đồ học trực tuyến [9] đã thiết kế thành công hai thuật toán ngẫu nhiên học mô hình LDA, đặt tên là ML-OPE và Online-OPE. Các tác giả trong [28] đã chứng minh được sự hiệu quả của hai thuật toán ML-OPE và Online-OPE so với các thuật toán học đương đại trong mô hình chủ đề. Chi tiết của ML-OPE và Online-OPE được trình bày trong Thuật toán 1.8 và Thuật toán 1.9.

1.7. Kết luận chương 1

Với lựa chọn đề tài nghiên cứu và đề xuất các phương pháp ngẫu nhiên giải bài toán cực đại hóa xác suất hậu nghiệm trong học máy, nên trong chương 1

Thuật toán 1.8 ML-OPE học LDA từ dữ liệu dòng/dữ liệu lớn

Đầu vào: Dữ liệu, tham số $K, \alpha, \tau > 0, \kappa \in (0.5, 1]$
Đầu ra: $\boldsymbol{\beta}$

1: Khởi tạo $\boldsymbol{\beta}^0$ ngẫu nhiên trong miền Δ_V
2: **for** $t = 1, 2, \ldots, \infty$ **do**
3: Lấy mini-batch $\mathcal{C}_t$ của tập các văn bản
4: Suy diễn bằng OPE cho mỗi văn bản $\mathbf{d} \in \mathcal{C}_t$ nhận được $\boldsymbol{\theta}_d$, cho bởi $\boldsymbol{\beta}^{t-1}$
5: Tính toán $\hat{\boldsymbol{\beta}}^t$ như sau:

$$\hat{\beta}^t_{kj} \propto \sum_{\mathbf{d} \in \mathcal{C}_t} d_j \theta_{dk}$$

6: Thiết lập tốc độ học $\rho_t = (t + \tau)^{-\kappa}$
7: Cập nhật $\boldsymbol{\beta}^t := (1 - \rho_t)\boldsymbol{\beta}^{t-1} + \rho_t \hat{\boldsymbol{\beta}}^t$
8: **end for**

Thuật toán 1.9 Online-OPE học LDA từ dữ liệu lớn

Đầu vào: Tập huấn luyện $\mathcal{C}$ với D văn bản, $K, \alpha, \eta, \tau > 0, \kappa \in (0.5, 1]$
Đầu ra: $\boldsymbol{\lambda}$

1: Khởi tạo $\boldsymbol{\lambda}^{\mathbf{0}}$ ngẫu nhiên
2: **for** $t = 1, 2, \ldots \infty$ **do**
3: Lấy mẫu nhỏ $\mathcal{C}_t$ bao gồm S văn bản,
4: Sử dụng thuật toán OPE để suy diễn hậu nghiệm cho mỗi văn bản $\mathbf{d} \in \mathcal{C}_t$, với biến toàn cục $\boldsymbol{\beta}^{t-1} \propto \boldsymbol{\lambda}^{t-1}$ trong bước trước, nhận được chủ đề hỗn hợp $\boldsymbol{\theta}_d$. Sau đó tính $\boldsymbol{\phi}_d$ như sau:

$$\phi_{djk} \propto \theta_{dk} \beta_{kj}$$

5: Với mỗi $k \in \{1, 2, \ldots, K\}$, biến toàn cục trung gian $\hat{\boldsymbol{\lambda}}_k$ cho $\mathcal{C}_t$ bởi

$$\hat{\lambda}_{kj} = \eta + \frac{D}{S} \sum_{\mathbf{d} \in \mathcal{C}_t} d_j \phi_{djk}$$

6: Cập nhật biến toàn cục bằng

$$\boldsymbol{\lambda}^t := (1 - \rho_t)\boldsymbol{\lambda}^{t-1} + \rho_t \hat{\boldsymbol{\lambda}}$$

trong đó $\rho_t = (t + \tau)^{-\kappa}$
7: **end for**

chúng tôi đã trình bày các kiến thức liên quan trực tiếp và gián tiếp đến đề tài. Cụ thể, chúng tôi trình bày khái quát về bài toán MAP và một số cách tiếp cận giải bài toán MAP, tiếp theo trình bày một số kiến thức cơ bản về tối ưu ngẫu nhiên giải bài toán tối ưu không lồi thường hay gặp trong học máy. Đồng thời các kiến thức cơ sở về mô hình đồ thị xác suất, các phương pháp suy diễn, mô hình chủ đề,... đã được chúng tôi đề cập đầy đủ trong chương này. Các nội dung tìm hiểu và được trình bày trong chương 1 là tiền đề cho các nghiên cứu về các thuật toán ngẫu nhiên giải bài toán MAP không lồi được đề xuất trong các chương tiếp theo. Một số kiến thức liên quan trong chương 1 được chúng tôi trình bày trong bài báo "How to make a machine learn continuously: a tutorial of the Bayesian approach" đăng trên kỷ yếu của hội thảo quốc tế Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications, SPIE, 2019.

Chương 2

NGẪU NHIÊN HÓA THUẬT TOÁN TỐI ƯU GIẢI BÀI TOÁN SUY DIỄN HẬU NGHIỆM TRONG MÔ HÌNH CHỦ ĐỀ

Trong chương này, nghiên cứu sinh đề xuất một nhóm thuật toán ngẫu nhiên theo cách tiếp cận tối ưu không lồi để giải bài toán suy diễn hậu nghiệm trong trong mô hình chủ đề, một minh họa điển hình cho bài toán MAP không lồi. Bài toán suy diễn hậu nghiệm trong trong mô hình chủ đề là một bài toán quan trọng khi phát triển mô hình chủ đề vào các ứng dụng thực tế. Đồng thời, nghiên cứu sinh đã mở rộng các thuật toán đề xuất này cho bài toán suy diễn Bayes và bài toán tối ưu không lồi tổng quát.

2.1. Giới thiệu

Trong khuôn khổ luận án nghiên cứu về các phương pháp ngẫu nhiên giải bài toán cực đại hóa xác suất hậu nghiệm không lồi, trong chương này, chúng tôi xem xét bài toán suy diễn hậu nghiệm trong mô hình chủ đề LDA [39]. Đây là một minh họa cho bài toán MAP không lồi trong các mô hình đồ thị xác suất, đối tượng nghiên cứu của luận án và được trình bày trong mục 1.3 chương 1. Như chúng tôi đã trình bày trong mục 1.4 đối với mô hình chủ đề, bài toán suy diễn hậu nghiệm đóng vai trò cốt lõi khi phát triển các mô hình chủ đề. Chúng tôi nhắc lại bài toán MAP đối với từng văn bản $\boldsymbol{d}$ trong mô hình chủ đề LDA chính là bài toán:

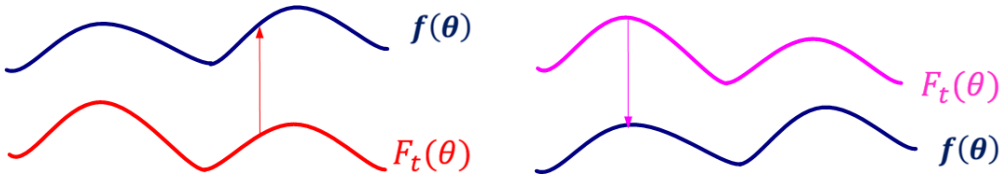
$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Delta_K} \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^K \log \theta_k \tag{2.1}$$

trong đó tham số $\alpha < 1$. Ký hiệu:

$$g_1(\boldsymbol{\theta}) := \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} \ , \ g_2(\boldsymbol{\theta}) := (\alpha - 1) \sum_{k=1}^K \log \theta_k$$

Khi đó bài toán (2.1) đưa về dạng:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Delta_K} [f(\boldsymbol{\theta}) = g_1(\boldsymbol{\theta}) + g_2(\boldsymbol{\theta})] \tag{2.2}$$



(a) $F_1(\boldsymbol{\theta}) = g_1(\boldsymbol{\theta})$ (b) $F_1(\boldsymbol{\theta}) = g_2(\boldsymbol{\theta})$

Hình 2.1: Hai trường hợp khởi tạo cho biên xấp xỉ ngẫu nhiên

Chúng tôi nhận thấy OPE [28] giải hiệu quả bài toán (2.2). Nghiên cứu các đặc điểm của OPE chúng tôi nhận thấy:

- Thành phần $g_1(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} < 0$ là log likelihood và $g_2(\boldsymbol{\theta}) = (\alpha - 1) \sum_{k=1}^K \log \theta_k > 0$ là log prior của văn bản $\mathbf{d}$.
- Hàm mục tiêu $f(\boldsymbol{\theta}) = g_1(\boldsymbol{\theta}) + g_2(\boldsymbol{\theta})$ bị kẹp giữa hai hàm g_1 và g_2, tức là $g_1(\boldsymbol{\theta}) < f(\boldsymbol{\theta}) < g_2(\boldsymbol{\theta})$.

Xét thuật toán OPE, trong lần lặp đầu tiên, nếu chọn $F_1 = g_1$ thì $F_1 < f$, dẫn đến dãy các hàm ngẫu nhiên xấp xỉ F_t tiến về f từ phía dưới, hay F_t là một cận dưới của f. Ngược lại, nếu chọn $F_1 = g_2$ trong lần lặp đầu tiên thì $F_1 > f$, do đó dãy các hàm ngẫu nhiên F_t tiến về f từ phía trên, hay nó là một cận trên của f (Chúng tôi mô tả minh họa trong Hình 2.1).

Điều này gợi ý cho chúng tôi nghĩ đến hai dãy hàm có cùng cách xây dựng nhưng xuất phát ban đầu khác nhau, cùng tiến đến hàm mục tiêu: một dãy bắt đầu từ $g_1(\boldsymbol{\theta})$ (phía dưới của hàm mục tiêu $f(\boldsymbol{\theta})$) và một dãy bắt đầu từ $g_2(\boldsymbol{\theta})$ (phía trên của hàm mục tiêu $f(\boldsymbol{\theta})$). Mỗi dãy hàm sẽ có tương ứng một dãy số tiến dần đến nghiệm $\boldsymbol{\theta}$ cần tìm. Ta tìm cách kết hợp hai dãy số này để được một dãy số tiến dần đến cực trị của hàm mục tiêu $f(\boldsymbol{\theta})$. Chỉ cần xây dựng hai dãy hàm ngẫu nhiên mà không phải cần nhiều hơn, vì hai dãy hàm số xuất phát từ bên trên và bên dưới hàm $f(\boldsymbol{\theta})$, bao lấy hàm $f(\boldsymbol{\theta})$ và tiến dần về $f(\boldsymbol{\theta})$. Nếu xây dựng nhiều hơn hai dãy hàm thì điểm xuất phát của các dãy hàm sẽ không nhiều riêng biệt như hai dãy trên và dưới, điều này không còn nhiều ý nghĩa trong việc xây dựng biên trên và biên dưới của hàm $f(\boldsymbol{\theta})$.

2.2. Đề xuất mới giải bài toán MAP trong mô hình chủ đề

Nhận thấy OPE là một thuật toán suy diễn hiệu quả cho bài toán MAP trong mô hình chủ đề. Hơn nữa, chúng tôi có thể cải tiến OPE theo hướng ngẫu nhiên hóa để nhận được các biến thể tốt hơn OPE ban đầu. Dựa trên ý tưởng của

OPE, chúng tôi đề xuất một số thuật toán cải tiến mới sẽ được trình bày trong mục này.

Xuất phát từ thành phần g_1, xây dựng dãy hàm $\{L_t\}$ như sau:

1. Khởi tạo $f_1^l := g_1$, khi đó $F_1 = g_1$;
2. Với $t = 2, 3, ...$ lựa chọn:
   (a) Lấy f_t^l có phân phối đều từ $\{g_1,\ g_2\}$;
   (b) $L_t := \frac{2}{t}\sum_{h=1}^{t} f_h^l$

Chuỗi hàm $\{L_t\}$ đóng vai trò là cận dưới của hàm mục tiêu đúng $f(\boldsymbol{\theta})$. Sử dụng phương pháp lặp, từ dãy hàm $\{L_t\}$ thu được dãy số $\{\boldsymbol{\theta}_t^l\},\ (t = 1, 2, ..)$ như sau:

$$\boldsymbol{\theta}_{t+1}^l := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^l - \boldsymbol{\theta}_t}{t}$$

trong đó $\boldsymbol{e}_t^l := \arg\max_{\boldsymbol{x}\in\Delta_K} \langle L_t'(\boldsymbol{\theta}_t), \boldsymbol{x}\rangle$.

Tương tự, xuất phát từ thành phần g_2, xây dựng dãy hàm $\{U_t\}$ như sau:

1. Khởi tạo $f_1^u := g_2$, khi đó $F_1 = g_2$;
2. Với $t = 2, 3, ...$ lựa chọn:
   (a) Lấy f_t^u có phân phối đều từ $\{g_1, g_2\}$;
   (b) $U_t := \frac{2}{t}\sum_{h=1}^{t} f_h^u$

Chuỗi hàm $\{U_t\}$ đóng vai trò là cận trên của hàm mục tiêu đúng $f(\boldsymbol{\theta})$. Bằng cách sử dụng phương pháp lặp, từ dãy hàm $\{U_t\}$, ta thu được dãy số $\{\boldsymbol{\theta}_t^u\},\ (t = 1, 2, ..)$ như sau:

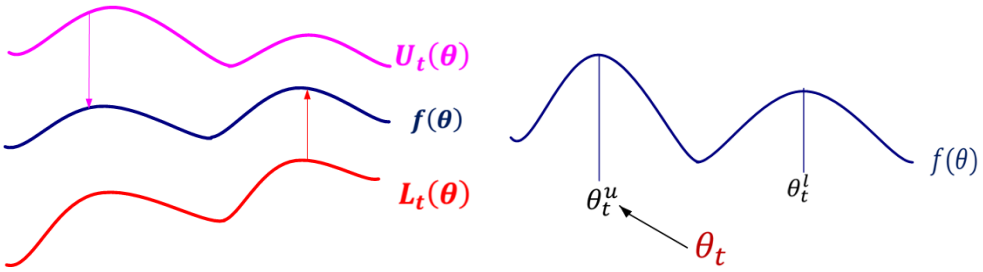
$$\boldsymbol{\theta}_{t+1}^u := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^l - \boldsymbol{\theta}_t}{t}$$

trong đó $\boldsymbol{e}_t^u := \arg\max_{\boldsymbol{x}\in\Delta_K} \langle U_t'(\boldsymbol{\theta}_t), \boldsymbol{x}\rangle$.
Như vậy có hai dãy hàm ngẫu nhiên $\{U_t\}$ và $\{L_t\}$ cùng xấp xỉ và tiến về hàm mục tiêu f (minh họa trong Hình 2.2a).

Để tăng tính ngẫu nhiên cho thuật toán đề xuất, tại mỗi bước lặp, nghiệm gần đúng $\boldsymbol{\theta}_t$ được chọn dựa vào hai dãy $\{\boldsymbol{\theta}_t^u\}$ và $\{\boldsymbol{\theta}_t^l\}$ bằng các phân phối xác suất thích hợp.

(1) **Cải tiến thứ nhất**: Sau khi xây dựng hai dãy $\{\boldsymbol{\theta}_t^u\}$ và $\{\boldsymbol{\theta}_t^l\}$, tiến hành lựa chọn nghiệm xấp xỉ $\boldsymbol{\theta}_t$ ở lần lặp thứ t theo phân phối đều từ hai nghiệm xấp xỉ trung gian $\{\boldsymbol{\theta}_t^u, \boldsymbol{\theta}_t^l\}$, tức là

$$P(\boldsymbol{\theta}_t = \boldsymbol{\theta}_t^u) = \frac{1}{2}\ ,\ P(\boldsymbol{\theta}_t = \boldsymbol{\theta}_t^l) = \frac{1}{2}$$



(a) Xây dựng biên trên và biên dưới của hàm mục tiêu $f(\boldsymbol{\theta})$

(b) Luôn lựa chọn điểm tốt hơn trong mỗi bước lặp

Hình 2.2: Mô tả ý tưởng cơ bản cải tiến thuật toán OPE.

Thuật toán 2.1 OPE1: Sự lựa chọn đều từ hai biên ngẫu nhiên

Đầu vào: Văn bản $\boldsymbol{d}$ và tham số mô hình $\{\boldsymbol{\beta}, \alpha\}$

Đầu ra: $\boldsymbol{\theta}^*$ là nghiệm cực đại hóa của hàm $f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^K \log \theta_k$

Khởi tạo $\boldsymbol{\theta}_1$ thuộc $\overline{\Delta}_K$
$f_1^l := \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}$; $f_1^u := (\alpha - 1) \sum_{k=1}^K \log \theta_k$
for $t = 2, 3, \ldots, \infty$ **do**
  Lấy f_t^u có phân phối đều từ $\{\sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}; (\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
  $U_t := \frac{2}{t} \sum_{h=1}^t f_h^u$
  $\boldsymbol{e}_t^u := \arg\max_{\boldsymbol{x} \in \overline{\Delta}_K} \langle U_t'(\boldsymbol{\theta}_t), \boldsymbol{x} \rangle$
  $\boldsymbol{\theta}_{t+1}^u := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}$
  Lấy f_t^l có phân phối đều từ $\{\sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}; (\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
  $L_t := \frac{2}{t} \sum_{h=1}^t f_h^l$
  $\boldsymbol{e}_t^l := \arg\max_{\boldsymbol{x} \in \overline{\Delta}_K} \langle L_t'(\boldsymbol{\theta}_t), \boldsymbol{x} \rangle$
  $\boldsymbol{\theta}_{t+1}^l := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^l - \boldsymbol{\theta}_t}{t}$
  Lấy $\boldsymbol{\theta}_{t+1}$ có phân phối đều từ $\{\boldsymbol{\theta}_{t+1}^u, \boldsymbol{\theta}_{t+1}^l\}$
end for

thu được thuật toán OPE1. Chi tiết của OPE1 được trình bày trong Thuật toán 2.1.

(2) **Cải tiến thứ hai**: Sau khi xây dựng hai dãy $\{\boldsymbol{\theta}_t^u\}$ và $\{\boldsymbol{\theta}_t^l\}$, tiến hành lựa chọn nghiệm xấp xỉ $\boldsymbol{\theta}_t$ ở lần lặp thứ t từ $\boldsymbol{\theta}_t^u$ và $\boldsymbol{\theta}_t^l$ dựa vào giá trị của hàm mục tiêu $f(\boldsymbol{\theta}_t^u)$ và $f(\boldsymbol{\theta}_t^l)$. Nghiệm $\boldsymbol{\theta}_t$ ở bước lặp thứ t được lựa chọn ngẫu nhiên từ $\boldsymbol{\theta}_t^u$ và $\boldsymbol{\theta}_t^l$ theo phân phối Bernoulli với xác suất q_t, tức là:

$$P(\boldsymbol{\theta}_t = \boldsymbol{\theta}_t^u) = q_t, \;\; P(\boldsymbol{\theta}_t = \boldsymbol{\theta}_t^l) = 1 - q_t$$

được tính bởi

$$q_t := \frac{\exp f(\boldsymbol{\theta}_t^u)}{\exp f(\boldsymbol{\theta}_t^u) + \exp f(\boldsymbol{\theta}_t^l)}$$

Chúng tôi thu được thuật toán cải tiến OPE2. Chi tiết của OPE2 được trình bày trong Thuật toán 2.2. Cách lựa chọn nghiệm xấp xỉ $\boldsymbol{\theta}_t$ trong mỗi bước lặp ở cải tiến OPE2 đã được làm mịn hơn so với biến thể OPE1 khi chúng tôi sử dụng nhiều thông tin của hàm mục tiêu f vào trong sự lựa chọn nghiệm $\boldsymbol{\theta}_t$ thay vì sử dụng phân phối đều như trong OPE1.

Thuật toán 2.2 OPE2: Làm mịn sự lựa chọn nghiệm từ hai biên ngẫu nhiên

Đầu vào: Văn bản d và tham số mô hình $\{\boldsymbol{\beta}, \alpha\}$

Đầu ra: $\boldsymbol{\theta}^*$ là nghiệm cực đại hóa của hàm $f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^K \log \theta_k$

Khởi tạo $\boldsymbol{\theta}_1$ thuộc $\overline{\Delta}_K$
$f_1^l := \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}$; $f_1^u := (\alpha - 1) \sum_{k=1}^K \log \theta_k$
for $t = 2, 3, \ldots, \infty$ **do**
  Lấy f_t^u có phân phối đều từ $\{\sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}; (\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
  $U_t := \frac{2}{t} \sum_{h=1}^t f_h^u$
  $\boldsymbol{e}_t^u := \arg\max_{\boldsymbol{x} \in \overline{\Delta}_K} \langle U_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} \rangle$
  $\boldsymbol{\theta}_{t+1}^u := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}$
  Lấy f_t^l có phân phối đều từ $\{\sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}; (\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
  $L_t := \frac{2}{t} \sum_{h=1}^t f_h^l$
  $\boldsymbol{e}_t^l := \arg\max_{\boldsymbol{x} \in \overline{\Delta}_K} \langle L_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} \rangle$
  $\boldsymbol{\theta}_{t+1}^l := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^l - \boldsymbol{\theta}_t}{t}$
  Lấy $\boldsymbol{\theta}_{t+1}$ theo phân phối xác suất $\{P(\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_{t+1}^u) = q_t,\ P(\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_{t+1}^l) = 1 - q_t\}$ trong đó xác suất q_t được xác định bởi

$$q_t := \frac{\exp f(\boldsymbol{\theta}_{t+1}^u)}{\exp f(\boldsymbol{\theta}_{t+1}^u) + \exp f(\boldsymbol{\theta}_{t+1}^l)}$$

end for

(3) **Cải tiến thứ ba:** Sau khi xây dựng hai dãy $\{\boldsymbol{\theta}_t^u\}$ và $\{\boldsymbol{\theta}_t^l\}$, chúng tôi tiến hành lựa chọn nghiệm xấp xỉ ở bước lặp t là:

$$\boldsymbol{\theta}_t := \arg \max_{\boldsymbol{\theta} \in \{\boldsymbol{\theta}_t^u, \boldsymbol{\theta}_t^l\}} f(\boldsymbol{\theta})$$

thu được thuật toán OPE3. Chi tiết của OPE3 được trình bày trong Thuật toán 2.3. Ý tưởng của lựa chọn nghiệm trong OPE3 dựa trên cách tiếp cận

Thuật toán 2.3 OPE3: Luôn lựa chọn nghiệm tốt hơn trong mỗi bước lặp

Đầu vào: văn bản $\boldsymbol{d}$ và tham số mô hình $\{\boldsymbol{\beta}, \alpha\}$

Đầu ra: $\boldsymbol{\theta}^*$ là nghiệm cực đại hóa của hàm $f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^K \log \theta_k$

Khởi tạo $\boldsymbol{\theta}_1$ thuộc $\overline{\Delta}_K$
$f_1^l := \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}$;$f_1^u := (\alpha - 1) \sum_{k=1}^K \log \theta_k$
for $t = 2, 3, .., \infty$ **do**
  Lấy f_t^u có phân phối đều từ $\{\sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}$; $(\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
  $U_t := \frac{2}{t} \sum_{h=1}^t f_h^u$
  $\boldsymbol{e}_t^u := \arg\max_{\boldsymbol{x} \in \overline{\Delta}_K} \langle U_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} \rangle$
  $\boldsymbol{\theta}_{t+1}^u := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}$
  Lấy f_t^l có phân phối đều từ $\{\sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}$; $(\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
  $L_t := \frac{2}{t} \sum_{h=1}^t f_h^l$
  $\boldsymbol{e}_t^l := \arg\max_{\boldsymbol{x} \in \overline{\Delta}_K} \langle L_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} \rangle$
  $\boldsymbol{\theta}_{t+1}^l := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^l - \boldsymbol{\theta}_t}{t}$
  Lấy $\boldsymbol{\theta}_{t+1} := \arg\max_{\boldsymbol{\theta} \in \{\boldsymbol{\theta}_{t+1}^u, \boldsymbol{\theta}_{t+1}^l\}} f(\boldsymbol{\theta})$
end for

tham lam. Ở mỗi lần lặp, chúng tôi luôn so sánh hai giá trị $f(\boldsymbol{\theta}_t^u)$ và $f(\boldsymbol{\theta}_t^l)$,

sau đó lấy nghiệm xấp xỉ làm cho hàm mục tiêu $f(\boldsymbol{\theta})$ đạt giá trị lớn nhất:

$$\boldsymbol{\theta}_t := \arg \max_{\boldsymbol{\theta} \in \{\boldsymbol{\theta}_t^u, \boldsymbol{\theta}_t^l\}} f(\boldsymbol{\theta})$$

có nghĩa là, nếu $f(\boldsymbol{\theta}_t^u) > f(\boldsymbol{\theta}_t^l)$ thì lựa chọn $\boldsymbol{\theta}_t := \boldsymbol{\theta}_t^u$, ngược lại chọn $\boldsymbol{\theta}_t := \boldsymbol{\theta}_t^l$.

(4) **Cải tiến thứ tư:** Như chúng tôi đã trình bày, các cải tiến OPE1, OPE2 và OPE3 sử dụng hai biên ngẫu nhiên từ phân phối đều kết hợp với lựa chọn nghiệm xấp xỉ trong mỗi bước lặp dựa vào hai dãy số $\{\boldsymbol{\theta}_t^u\}$ và $\{\boldsymbol{\theta}_t^l\}$. Chúng tôi có một ý tưởng khác, đó là xấp xỉ hàm mục tiêu đúng $f(\boldsymbol{\theta})$ bởi một hàm xấp xỉ ngẫu nhiên $F_t(\boldsymbol{\theta})$, trong đó $F_t(\boldsymbol{\theta})$ là tổ hợp tuyến tính của hai biên ngẫu nhiên U_t và L_t với tham số tổ hợp $\nu \in (0, 1)$ được lựa chọn thích hợp:

$$F_t(\boldsymbol{\theta}) := \nu U_t(\boldsymbol{\theta}) + (1 - \nu) L_t(\boldsymbol{\theta})$$

và tiến hành tìm nghiệm $\boldsymbol{\theta}_t$ tương tự như OPE. Chúng tôi thu được OPE4 và được trình bày chi tiết trong Thuật toán 2.4.

Thuật toán 2.4 OPE4: Sử dụng tổ hợp tuyến tính của các biên ngẫu nhiên

Đầu vào: Văn bản $\mathbf{d}$, tham số tổ hợp $\nu \in (0, 1)$ và tham số mô hình $\{\boldsymbol{\beta}, \alpha\}$
Đầu ra: $\boldsymbol{\theta}^*$ là nghiệm cực đại hóa của hàm $f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^K \log \theta_k$

Khởi tạo $\boldsymbol{\theta}_1$ thuộc $\overline{\Delta}_K$
$f_1^l := \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}$; $f_1^u := (\alpha - 1) \sum_{k=1}^K \log \theta_k$
for $t = 2, 3, .., \infty$ **do**
  Lấy f_t^u theo phân phối đều từ tập $\{\sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj};\ (\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
  $U_t := \frac{2}{t} \sum_{h=1}^t f_h^u$
  Lấy f_t^l theo phân phối đều từ tập $\{\sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj};\ (\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
  $L_t := \frac{2}{t} \sum_{h=1}^t f_h^l$
  Lập tổ hợp tuyến tính $F_t := \nu U_t + (1 - \nu) L_t$
  $\boldsymbol{e}_t := \arg\max_{\boldsymbol{x} \in \overline{\Delta}_K} < F_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} >$
  $\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}$
end for

2.3. Các thuật toán học ngẫu nhiên cho mô hình LDA

Tác giả trong [28] đã sử dụng OPE để thiết kế các thuật toán học nhanh mô hình chủ đề: thuật toán ML-OPE cho phép học mô hình LDA từ dữ liệu dòng/dữ liệu lớn, còn Online-OPE học mô hình LDA từ các tập dữ liệu lớn. Các thuật toán này sử dụng OPE để thực hiện suy diễn hậu nghiệm cho các văn bản riêng lẻ và sơ đồ trực tuyến [101, 32] để suy diễn ra các biến toàn cục (chủ đề). Do đó, bản chất ngẫu nhiên xuất hiện trong cả hai giai đoạn suy diễn cục bộ và toàn cục. Lưu ý rằng suy diễn MAP cho các biến cục bộ của OPE đã được chứng minh có đảm bảo lý thuyết về tốc độ hội tụ nhanh. Do đó, chúng

tôi tiến hành thay đổi thuật toán lõi suy diễn OPE bằng các cải tiến mới như OPE1, OPE2, OPE3 và OPE4, sau đó đưa vào trong thuật toán học ML-OPE và Online-OPE [28] và thu được 8 thuật toán học ngẫu nhiên mới để học mô hình LDA. Đó là: ML-OPE1, ML-OPE2, ML-OPE3, ML-OPE4, Online-OPE1, Online-OPE2, Online-OPE3 và Online-OPE4. Chi tiết các thuật toán được mô tả trong Thuật toán 2.5 và Thuật toán 2.6.

Thuật toán 2.5 ML-OPE1(2,3,4) học LDA từ dữ liệu dòng/dữ liệu lớn

Đầu vào: Tham số $K, \alpha, \tau > 0, \kappa \in (0.5, 1]$
Đầu ra: $\boldsymbol{\beta}$

1: Khởi tạo $\boldsymbol{\beta}^0$ ngẫu nhiên trong miền Δ_V
2: **for** $t = 1, 2, \ldots \infty$ **do**
3: Lấy tập nhỏ $\mathcal{C}_t$ của tập các văn bản
4: Thực hiện suy diễn bằng OPE1 (OPE2, OPE3, OPE4) cho mỗi văn bản $\mathbf{d} \in \mathcal{C}_t$ nhận được $\boldsymbol{\theta}_d$ cho bởi $\boldsymbol{\beta}^{t-1}$.
5: Tính toán chủ đề $\hat{\boldsymbol{\beta}}^t$ như sau: $\hat{\beta}^t_{kj} \propto \sum_{\mathbf{d} \in \mathcal{C}_t} d_j \theta_{dk}$
6: Thiết lập tốc độ học $\rho_t = (t + \tau)^{-\kappa}$, cập nhật $\boldsymbol{\beta}^t := (1 - \rho_t)\boldsymbol{\beta}^{t-1} + \rho_t \hat{\boldsymbol{\beta}}^t$
7: **end for**

Thuật toán 2.6 Online-OPE1(2,3,4) học LDA từ dữ liệu lớn

Đầu vào: Tập dữ liệu huấn luyện $\mathcal{C}$ với D văn bản, $K, \alpha, \eta, \tau > 0, \kappa \in (0.5, 1]$
Đầu ra: $\boldsymbol{\lambda}$

1: Khởi tạo $\boldsymbol{\lambda}^{\mathbf{0}}$ ngẫu nhiên
2: **for** $t = 1, 2, \ldots, \infty$ **do**
3: Lấy mẫu nhỏ $\mathcal{C}_t$ gồm S văn bản.
4: Sử dụng OPE1 (OPE2, OPE3, OPE4) để suy diễn hậu nghiệm cho mỗi văn bản $\mathbf{d} \in \mathcal{C}_t$, cho bởi biến toàn cục $\boldsymbol{\beta}^{t-1} \propto \boldsymbol{\lambda}^{t-1}$ trong bước trước đó để nhận được chủ đề hỗn hợp $\boldsymbol{\theta}_d$. Tính toán $\boldsymbol{\phi}_d$ theo:

$$\phi_{djk} \propto \theta_{dk} \beta_{kj}$$

5: Với mỗi $k \in \{1, 2, .., K\}$, biến toàn cục $\hat{\boldsymbol{\lambda}}_k$ cho $\mathcal{C}_t$ bởi

$$\hat{\lambda}_{kj} = \eta + \frac{D}{S} \sum_{\mathbf{d} \in \mathcal{C}_t} d_j \phi_{djk}$$

6: Với $\rho_t = (t + \tau)^{-\kappa}$, cập nhật biến toàn cục

$$\boldsymbol{\lambda}^t := (1 - \rho_t)\boldsymbol{\lambda}^{t-1} + \rho_t \hat{\boldsymbol{\lambda}}$$

7: **end for**

2.4. Đánh giá thực nghiệm

Trong phần này, chúng tôi tiến hành các thực nghiệm để kiểm tra hiệu quả thực tế của các cải tiến mới của chúng tôi để giải bài toán MAP trong LDA. Vì OPE, OPE1, OPE2, OPE3 và OPE4 có thể đóng vai trò là thuật toán suy diễn cốt lõi trong các phương pháp học với mô hình LDA, do đó sự hiệu quả của các biến thể mới được đánh giá thông qua đánh giá sự hiệu quả của các phương

pháp học tương ứng.

2.4.1. Các bộ dữ liệu thực nghiệm

Chúng tôi tiến hành thực nghiệm cho các cải tiến trên hai bộ dữ liệu lớn: bộ New York Times (NYT) bao gồm 300.000 bài tin tức và bộ PubMed (PUB) bao gồm 330.000 bài báo từ trung tâm PubMed[1]. Chi tiết của hai bộ dữ liệu được mô tả trong Bảng 2.1.

Bộ dữ liệu	**Số văn bản**	**Số thuật ngữ**	**Số văn bản huấn luyện**	**Số văn bản kiểm tra**	**Độ dài văn bản TB**
New York Times	300,000	141,444	290,000	10,000	325.13
PubMed	330,000	100,000	320,000	10,000	65.12

Bảng 2.1: Hai bộ dữ liệu thực nghiệm

2.4.2. Độ đo đánh giá thực nghiệm

Chúng tôi đã sử dụng hai độ đo thường được dùng trong mô hình chủ đề, đó là *Log Predictive Probability* (LPP) [43] và *Normalised Pointwise Mutual Information* (NPMI) [102]. LPP đo lường tính dự đoán và khái quát hóa của một mô hình đối với dữ liệu mới, trong khi NPMI đánh giá chất lượng ngữ nghĩa của một chủ đề riêng lẻ. Độ đo LPP và NPMI càng cao càng tốt, thể hiện thuật toán học càng hiệu quả. Chi tiết về cách tính các độ đo này được trình bày trong Phụ lục A và B.

2.4.3. Kết quả thực nghiệm

Bài báo [28] đã phân tích rất kỹ lưỡng và chứng minh được thuật toán suy diễn OPE thường có hiệu quả tốt hơn một số phương pháp suy diễn đương đại như VB, CVB0 hay CGS trong quá trình suy diễn. Do đó, trong phần này, chúng tôi tiến hành thực nghiệm đánh giá sự hiệu quả của các thuật toán mới đề xuất thông qua so sánh với OPE. Để đảm bảo các kết quả so sánh là công bằng, nên trong thực nghiệm các tham số tự do được thiết lập như trong bài báo [28] cho mỗi bộ dữ liệu và mỗi phương pháp học, cụ thể như sau:

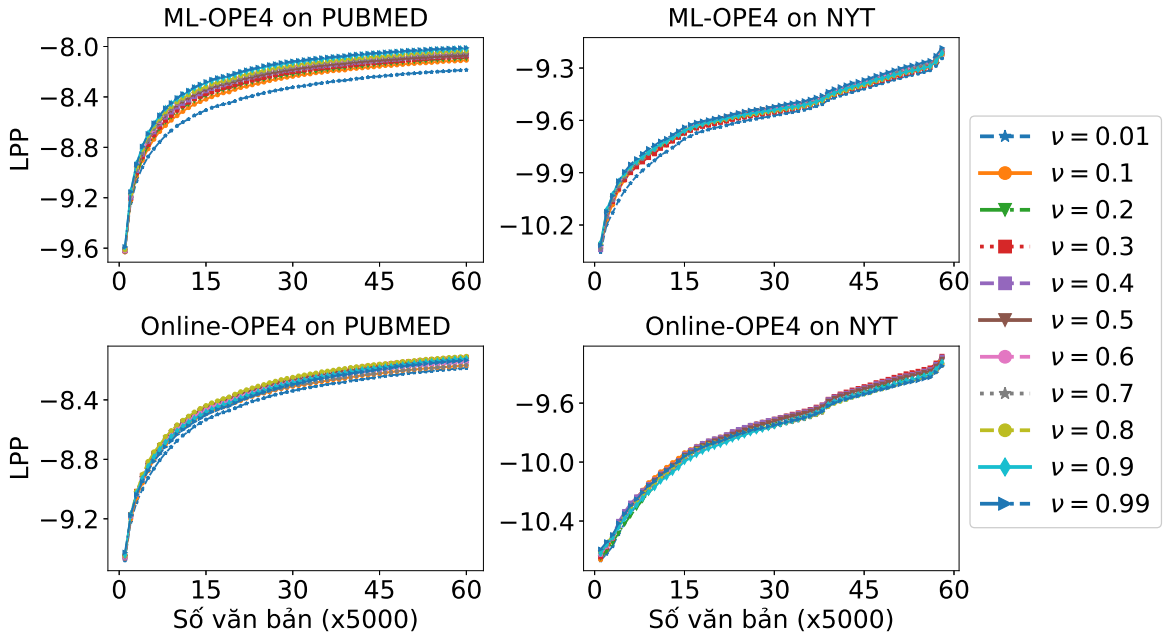
- **Tham số mô hình:** Thiết lập số chủ đề $K = 100$, tham số Dirichlet $\alpha = \frac{1}{K}$ và siêu tham số $\eta = \frac{1}{K}$. Các tham số này thường được sử dụng trong các mô hình chủ đề.

[1]Các bộ dữ liệu được lấy từ http://archive.ics.uci.edu/ml/datasets

- **Tham số suy diễn:** Lựa chọn số bước lặp của thuật toán suy diễn $T = 50$. Ngoài ra, khảo sát sự ảnh hưởng của số lần lặp T đến các thuật toán suy diễn và thuật toán học, chúng tôi cũng tiến hành thực nghiệm với các giá trị khác nhau của $T \in \{20, 30, 40, 50, 100\}$. Trong thuật toán OPE4, chúng tôi có khảo sát tham số tổ hợp tuyến tính ν nhận các giá trị rời rạc trong $\{0.01, 0.10, 0.20, \ldots, 0.90, 0.99\}$.

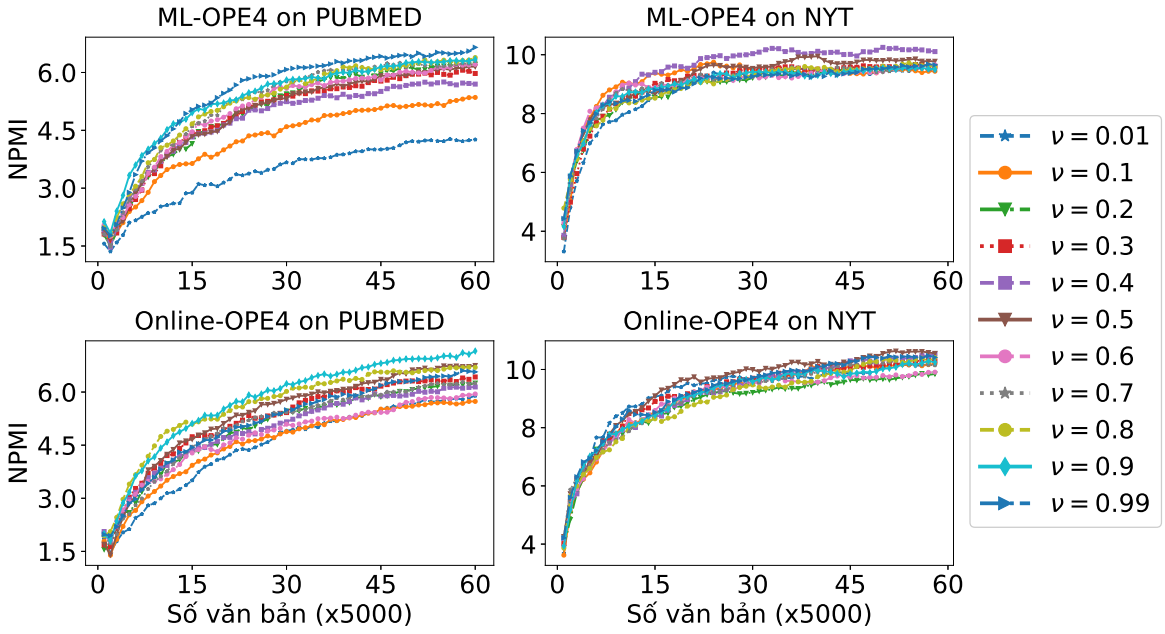
- **Tham số học:** Lựa chọn kích thước mini-batch $S = |C_t| = 5000$, thiết lập siêu tham số $\kappa = 0.9$ và $\tau = 1$ thích nghi tốt cho các phương pháp suy luận hiện có.

Để tăng độ tin cậy và sự ổn định trong các kết quả đánh giá, với mỗi bộ dữ liệu, chúng tôi tiến hành thực nghiệm cho mỗi phương pháp học trên một bộ dữ liệu 5 lần và lấy kết quả trung bình. Thuật toán OPE4 có sự có mặt của tham số tổ hợp ν và hiệu quả của OPE4 phụ thuộc vào giá trị tham số ν được chọn. Để xem xét ảnh hưởng của tham số tổ hợp hai biên ngẫu nhiên ν trong OPE4, chúng tôi đã thay đổi giá trị của ν trong $\{0.01, 0.10, 0.20, .., 0.90, 0.99\}$ và ghi lại kết quả thực hiện của OPE4 tương ứng với các giá trị của tham số ν được lựa chọn. Chi tiết kết quả thực nghiệm ML-OPE4 và Online-OPE4 với các giá trị khác nhau của ν trong khoảng từ 0 đến 1 được mô tả trong Hình 2.3 và Hình 2.4. Thông qua kết quả thực nghiệm trong Hình 2.3 và Hình 2.4 cho thấy chất



Hình 2.3: Kết quả thực hiện của OPE4 với tham số ν được lựa chọn khác nhau trên độ đo LPP.

lượng của OPE4 có thể tốt hơn khi lựa chọn được tham số tổ hợp ν phù hợp.



Hình 2.4: Kết quả thực hiện của OPE4 với tham số ν được lựa chọn khác nhau trên độ đo NPMI.

Chúng tôi nhận thấy thuật toán OPE4 phù hợp với tham số ν có xu hướng gần giá trị 0.5 đối với bộ New York Times hay gần giá trị 1 với bộ PubMed. Các giá trị phù hợp của tham số tổ hợp ν được trình bày trong Bảng 2.2.

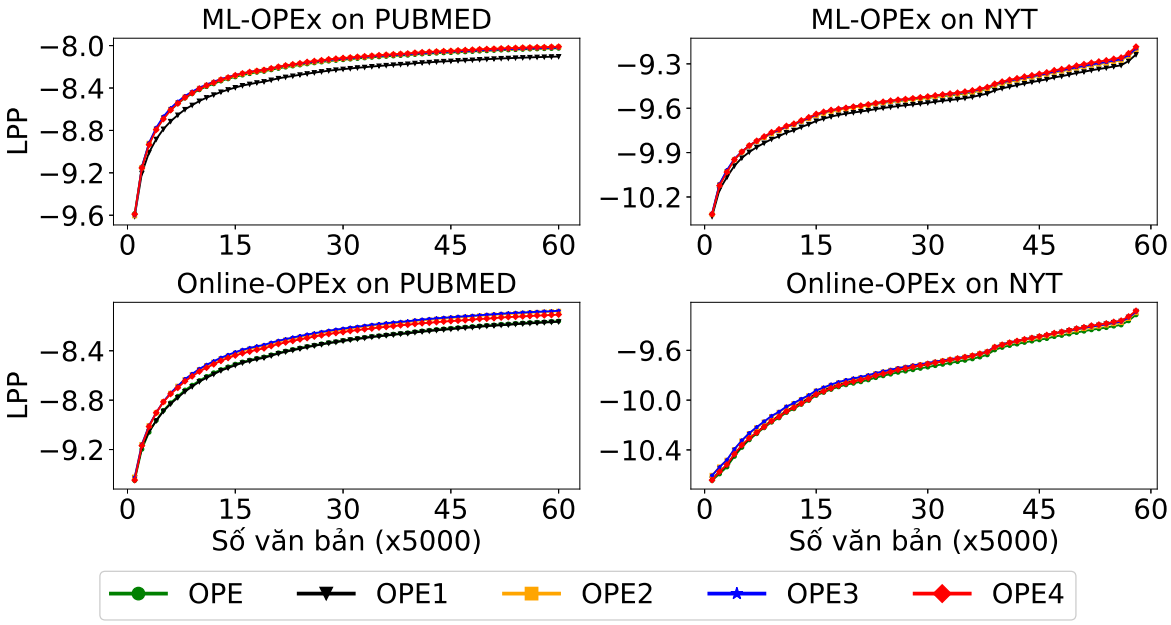
Phương pháp học	**Độ đo**	**Bộ New York Times**	**Bộ PubMed**
ML-OPE4	LPP	$\nu = 0.6$	$\nu = 0.99$
ML-OPE4	NPMI	$\nu = 0.4$	$\nu = 0.99$
Online-OPE4	LPP	$\nu = 0.3$	$\nu = 0.8$
Online-OPE4	NPMI	$\nu = 0.5$	$\nu = 0.9$

Bảng 2.2: Giá trị của tham số tổ hợp ν phù hợp nhất với từng phương pháp học trên các bộ dữ liệu khác nhau.

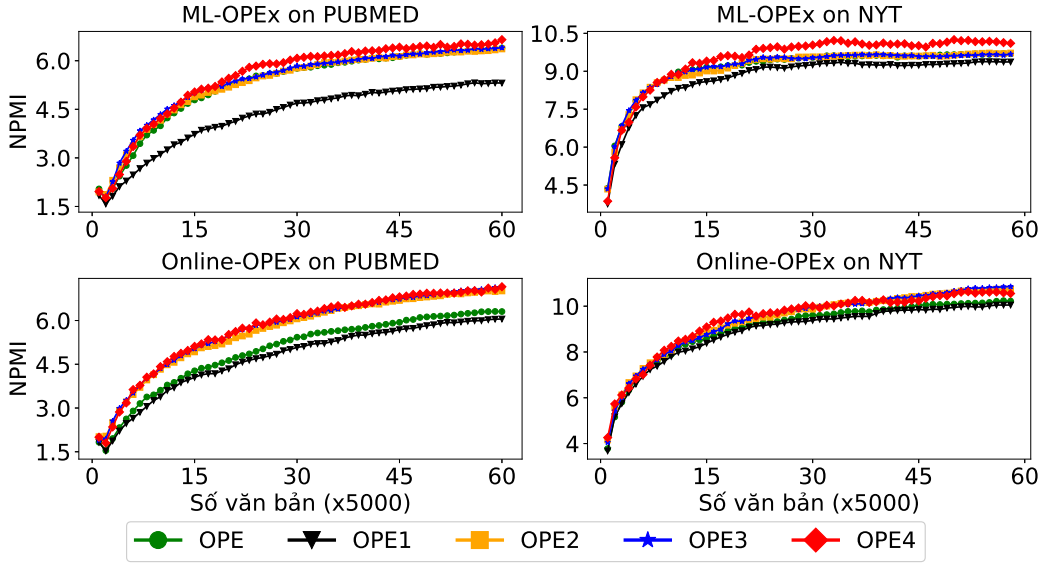
Chúng tôi tiến hành thực nghiệm các thuật toán mới đề xuất OPE1, OPE2, OPE3 và OPE4 (với giá trị tham số ν phù hợp được lựa chọn trong Bảng 2.2) và so sánh với thuật toán OPE được đề xuất trong [28]. Chi tiết kết quả được mô tả trong Hình 2.5 và Hình 2.6.

Các biến thể OPE1, OPE2, OPE3 và OPE4 nhằm tìm kiếm tham số $\boldsymbol{\theta}$ tối đa hóa hàm $f(\boldsymbol{\theta})$ trên một đơn hình bằng cách sử dụng hai biên ngẫu nhiên. Sau đó, kết quả của nó được sử dụng để cập nhật các tham số học của mô hình. Cách học tiếp cận theo thuật toán học ML-OPE cập nhật tham số $\boldsymbol{\beta}$ trực tiếp còn tiếp cận theo thuật toán Online-OPE lại cập nhật tham số biến phân $\boldsymbol{\lambda}$. Chất lượng của tham số $\boldsymbol{\theta}$ tìm được bởi các thuật toán suy diễn ảnh hưởng trực tiếp đến chất lượng tham số $\boldsymbol{\beta}$ và $\boldsymbol{\lambda}$ trong mô hình LDA.

Hình 2.5 cho thấy OPE1 và OPE2 hoạt động kém hơn hơn các thuật toán



Hình 2.5: Kết quả của các thuật toán mới so sánh với OPE thông qua độ đo LPP. Độ đo càng cao càng tốt. Chúng tôi thấy rằng một số thuật toán mới đảm bảo tốt hoặc thậm chí tốt hơn OPE.



Hình 2.6: Kết quả của các thuật toán mới so sánh với OPE trên độ đo NPMI. Độ đo càng cao càng tốt. Chúng tôi thấy rằng một số thuật toán mới đảm bảo tốt, thậm chí tốt hơn OPE.

còn lại. Cách hoạt động của OPE1 và OPE2 không làm tăng tính ngẫu nhiên của các xấp xỉ. Ở mỗi lần lặp, cả OPE1 và OPE2 đều ngẫu nhiên chọn nghiệm $\boldsymbol{\theta}_t$ một trong hai giá trị trong $\boldsymbol{\theta}_t^u$ và $\boldsymbol{\theta}_t^l$. Như vậy, đối với các lần lặp liên tiếp, nó không đảm bảo là chọn được giá trị nghiệm làm cho hàm mục tiêu f tăng. OPE3 đã khắc phục vấn đề này. Thuật toán OPE3 luôn lựa chọn nghiệm $\boldsymbol{\theta}_t$ sao cho giá trị của hàm mục tiêu f tăng. Tức là, chất lượng của tham số $\boldsymbol{\theta}$ được tốt hơn, nên chất lượng của tham số $\boldsymbol{\beta}$ trong thuật toán học được tốt hơn. Xác suất

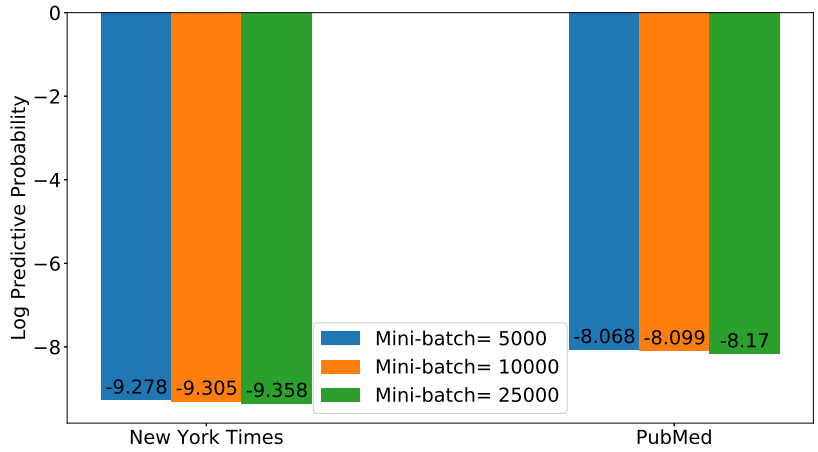
dự đoán thu được bởi OPE3 cao hơn kết quả tương ứng của OPE1 hoặc OPE2. Tương tự như OPE3, OPE4 với tham số tổ hợp ν phù hợp đã cho kết quả tốt hơn các biến thể khác. Ngoài ra, theo kết quả mô tả ở Hình 2.5, ta thấy độ đo LPP của các phương pháp khác biệt không nhiều. Bởi vì LPP phụ thuộc vào chất lượng của tham số $\boldsymbol{\beta}$ của phương pháp học ML-OPE và Online-OPE. Kết quả mô tả trong Hình 2.5 thể hiện chất lượng của tham số $\boldsymbol{\theta}$ chưa được cải thiện nhiều trong quá trình suy diễn. Ngược lại, Hình 2.6 cho thấy độ đo NPMI được cải thiện đáng kể bởi các biến thể OPE mới này. Chúng tôi phát hiện ra rằng OPE1 thu được kết quả kém nhất, OPE2 và OPE3 tốt hơn OPE, còn OPE4 (với tham số tổ hợp ν phù hợp) cho kết quả tốt nhất. NPMI được tính trực tiếp từ tham số $\boldsymbol{\theta}$ học được.

Dễ dàng nhận thấy chất lượng của tham số $\boldsymbol{\theta}$ được cải thiện đáng kể với cách xây dựng các xấp xỉ mới của hàm mục tiêu f từ OPE2 và OPE3, đặc biệt là OPE3. OPE4 được chỉ ra là hiệu quả hơn khi tham số tổ hợp ν được lựa chọn phù hợp nhất. Bằng cách thêm tham số ν thích hợp, OPE4 đã tăng chất lượng của mô hình bởi vì trong lý thuyết học máy mô hình càng phức tạp thì độ chính xác mà nó đạt được càng cao.

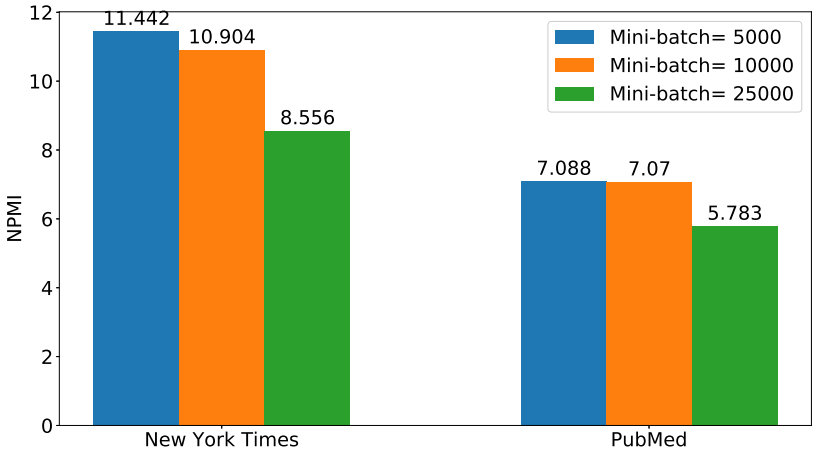
Ngoài ra chúng tôi cũng tiến hành một số thực nghiệm để khảo sát sự ảnh hưởng của cách chia tập dữ liệu (kích thước các mini-batch), sự ảnh hưởng của số bước lặp T trong các thuật toán suy diễn, cũng như so sánh thời gian thực hiện của các thuật toán. Chúng tôi thiết lập tham số theo nghiên cứu [28]: Số chủ đề $K = 100$, tham số Dirichlet $\alpha = \frac{1}{K}$ và $\eta = \frac{1}{K}$; tham số học $\kappa = 0.9$ và $\tau = 1$. Chúng tôi sử dụng thuật toán học Online-OPE3 để thực nghiệm khảo sát sự thay đổi của kích thước mini-batch $|C_t|$ và số bước lặp T của thuật toán suy diễn OPE3.

Chúng tôi khảo sát ảnh hưởng của việc lựa chọn kích thước mini-batch, chúng tôi tiến hành thực nghiệm Online-OPE3 trên hai bộ dữ liệu New York Times và PubMed với lựa chọn kích thước mini-batch lần lượt là 5000, 10000 và 25000. Chi tiết kết quả được mô tả trong Hình 2.7 và Hình 2.8.

Theo kết quả mô tả trong Hình 2.7 và Hình 2.8, chúng tôi nhận thấy thuật toán Online-OPE3 thực hiện với cách chia bộ dữ liệu theo kích thước mini-batch là 5000 cho kết quả tốt hơn trường hợp kích thước mini-batch là 10000 và 25000. Điều đó hoàn toàn phù hợp với tư tưởng của các thuật toán học "online" là làm việc ngẫu nhiên trên các mẫu nhỏ của bộ dữ liệu lớn, tức là người ta chia bộ dữ liệu thành nhiều mini-batch nhỏ và tiến hành huấn luyện trên từng mẫu nhỏ.



Hình 2.7: Kết quả độ đo LPP của thuật toán học Online-OPE3 trên hai bộ dữ liệu New York Times và PubMed với các cách chia kích thước mini-batch khác nhau. Độ đo càng cao càng tốt.

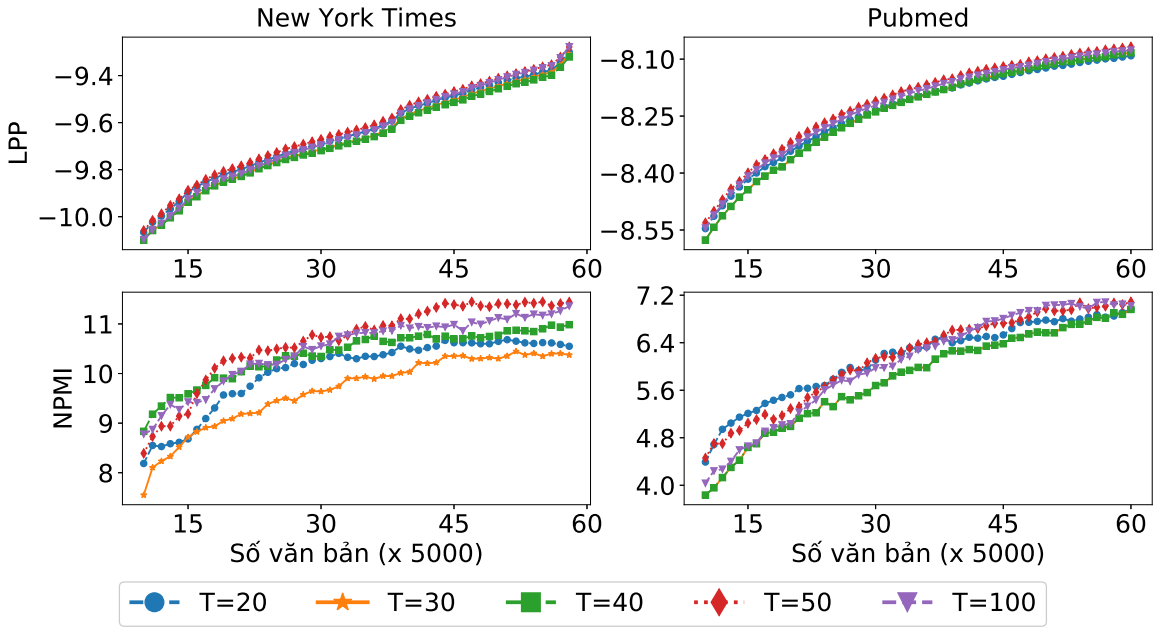


Hình 2.8: Kết quả độ đo NPMI của thuật toán học Online-OPE3 trên hai bộ dữ liệu New York Times và PubMed với các cách chia kích thước mini-batch khác nhau. Độ đo càng cao càng tốt.

Còn khi chia bộ dữ liệu thành các mini-batch kích thước lớn, cách thức hoạt động của thuật toán có xu hướng theo cách học "batch" cổ điển kém hiệu quả [9, 99].

Theo hiểu biết của chúng tôi, chất lượng của nghiệm xấp xỉ thu được phụ thuộc vào số bước lặp T của thuật toán suy diễn. Tuy nhiên, nếu lựa chọn số bước lặp T quá lớn sẽ làm mất nhiều thời gian thực hiện, ngược lại khi T quá nhỏ sẽ làm giảm chất lượng nghiệm thu được. Chúng tôi tiến hành khảo sát số bước lặp $T \in \{20, 30, 40, 50, 100\}$ trong thuật toán suy diễn OPE3 thông qua thực nghiệm thuật toán học Online-OPE3 trên hai bộ dữ liệu New York Times và PubMed. Kết quả được chi tiết trong Hình 2.9.

Theo Hình 2.9, chúng tôi nhận thấy độ đo LPP và NPMI của thuật toán học Online-OPE3 có biến động khi thay đổi số bước lặp T trong thuật toán suy diễn OPE3, nhưng sự biến động trên các độ đo (đặc biệt là LPP) không quá lớn.



Hình 2.9: Kết quả độ đo LPP và NPMI của thuật toán học Online-OPE3 trên hai bộ dữ liệu New York Times và PubMed khi thay đổi số bước lặp T trong thuật toán suy diễn OPE3. Độ đo càng cao càng tốt.

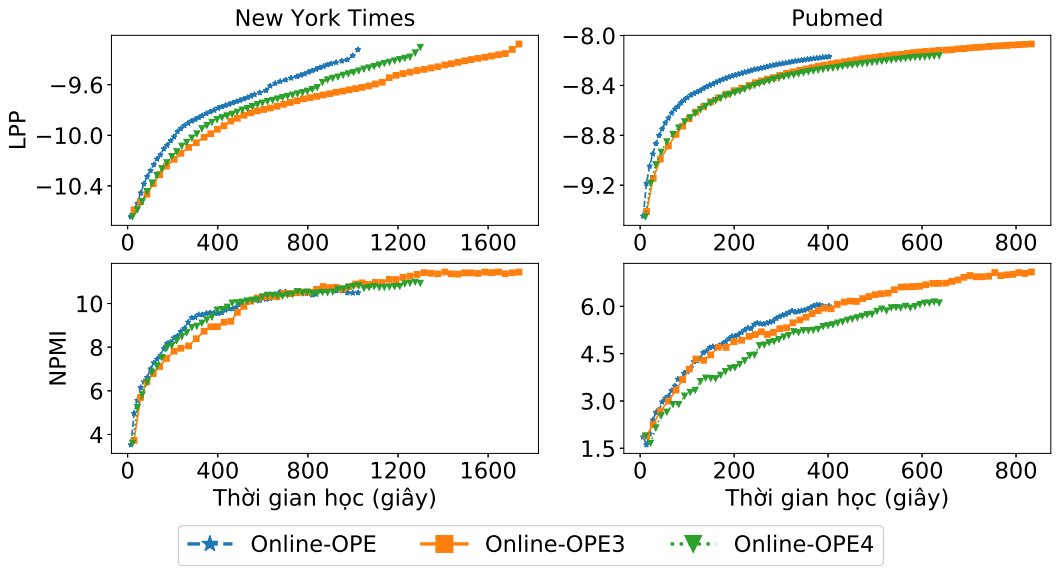
Đồng thời khi tăng số bước lặp T (ví dụ $T = 100$) cũng không phải luôn đảm bảo tìm được nghiệm $\boldsymbol{\theta}$ phù hợp nhất. Có thể lý giải cho điều này là do các thuật toán OPE, OPE3 và OPE4 có tốc độ hội tụ nhanh, nên không cần thực hiện quá nhiều bước lặp đã thu được nghiệm đủ tốt và ổn định. Theo Hình 2.9, chúng tôi thấy lựa chọn $T = 50$ đã đảm bảo kết quả các độ đo tốt của thuật toán học Online-OPE3 mà không tốn quá nhiều bước lặp.

Chúng tôi cũng tiến hành đo thời gian thực hiện thuật toán học: tính tổng thời gian thực hiện bước E và bước M cho mỗi thuật toán học Online-OPE, Online-OPE3 và Online-OPE4. Kết quả chi tiết được mô tả trong Hình 2.10 và Bảng 2.3.

Bộ dữ liệu	**Phương pháp học**	**Thời gian**	**Độ đo LPP**	**Độ đo NPMI**
New York Times	Online-OPE	1022.21	-9.32	10.50
	Online-OPE3	1737.18	**-9.28**	**11.44**
	Online-OPE4	1298.88	-9.30	10.93
PubMed	Online-OPE	402.23	-8.17	6.01
	Online-OPE3	832.69	**-8.07**	**7.09**
	Online-OPE4	636.45	-8.15	6.11

Bảng 2.3: Bảng thống kê thời gian thực hiện và độ đo của thuật toán học Online-OPE, Online-OPE3 và Online-OPE4 ($\nu = 0.3$) khi thực nghiệm trên hai bộ dữ liệu New York Times và PubMed.

Chúng tôi thực nghiệm trên hai bộ dữ liệu New York Times và PubMed với số lượng văn bản được lấy là tương đương nhau, nhưng độ dài của các văn bản trong bộ New York Times (trung bình khoảng 325 từ trên một văn bản) lớn



Hình 2.10: Kết quả độ đo LPP và NPMI tương ứng với thời gian thực hiện thuật toán học Online-OPE, Online-OPE3 và Online-OPE4 ($\nu = 0.3$) trên hai bộ dữ liệu New York Times và PubMed.

hơn trong bộ PubMed (trung bình khoảng 65 từ trên một văn bản) rất nhiều nên thời gian thực hiện thuật toán trên New York Times nhiều hơn PubMed. Hơn nữa, trên cùng một bộ dữ liệu, thời gian thực hiện Online-OPE3 là cao nhất, cao hơn Online-OPE4 và Online-OPE là thấp nhất. Sở dĩ Online-OPE3 mất nhiều thời gian vì số phép toán tại mỗi vòng lặp của OPE3 thường gấp đôi của OPE. Tuy nhiên, tổng thời gian thực hiện các thuật toán học không lớn nên sự khác biệt đó không đáng kể và hoàn toàn có thể chấp nhận đánh đổi khi chất lượng độ đo LPP và NPMI của Online-OPE3 thu được tốt hơn hẳn so với Online-OPE.

2.5. Sự hội tụ của các thuật toán đề xuất

Từ các kết quả thực nghiệm ở trên, chúng tôi nhận thấy OPE3 và OPE4 hiệu quả hơn OPE với hai bộ dữ liệu thực nghiệm khi áp dụng vào thiết kế hai thuật toán học với LDA. Vì vậy, chúng tôi tập trung vào phân tích sự hội tụ của thuật toán OPE3 và OPE4.

Định lý 2.1 (Sự hội tụ của thuật toán OPE3)**.** *Xem xét hàm mục tiêu $f(\boldsymbol{\theta})$ trong bài toán* (2.1)*, cho trước văn bản $\mathbf{d}$, tham số $\boldsymbol{\beta}$ và α. Xét thuật toán OPE3, với xác suất 1, ta có:*

(i) Với $\boldsymbol{\theta} \in \Delta_K$, dãy biên $U_t(\boldsymbol{\theta})$ và $L_t(\boldsymbol{\theta})$ hội tụ tới $f(\boldsymbol{\theta})$ khi $t \to +\infty$;

(ii) Dãy nghiệm xấp xỉ $\{\boldsymbol{\theta}_t\}$ hội tụ tới điểm dừng/điểm cực trị địa phương của hàm mục tiêu $f(\boldsymbol{\theta})$ khi $t \to +\infty$.

Chứng minh. Từ bài toán (2.1), ta thấy hàm mục tiêu $f(\boldsymbol{\theta})$ là hàm không lồi. Tiêu chuẩn được sử dụng để phân tích hội tụ rất quan trọng trong tối ưu hóa không lồi. Đối với các bài toán tối ưu không lồi không ràng buộc, gradient của hàm mục tiêu $\|\nabla f(\boldsymbol{\theta})\|$ được dùng để đánh giá hội tụ, bởi vì $\|\nabla f(\boldsymbol{\theta})\| \to 0$ đưa đến hội tụ đến một điểm dừng. Tuy nhiên, tiêu chuẩn này không thể được sử dụng cho các bài toán tối ưu không lồi có ràng buộc. Thay vào đó, ta sử dụng tiêu chuẩn "Frank-Wolfe gap" trong [72]. Ký hiệu

$$g_1(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj} \ , \ g_2(\boldsymbol{\theta}) = (\alpha - 1) \sum_{k=1}^{K} \log \theta_k$$

Đầu tiên, ta xem xét dãy $\{U_t\}$. Gọi a_t và b_t là số lần lấy được thành phần g_1 và g_2 tương ứng sau t lần lặp để xây dựng dãy $\{U_t\}$. Chúng ta thấy $a_t + b_t = t$. Ký hiệu $S_t = a_t - b_t$. Chúng ta có

$$U_t = \frac{2}{t}(a_t g_1 + b_t g_2) \tag{2.3}$$

$$U_t - f = \frac{S_t}{t}(g_1 - g_2) \tag{2.4}$$

$$U_t' - f' = \frac{S_t}{t}(g_1' - g_2') \tag{2.5}$$

Vì f_t^u được chọn theo phân phối đều từ $\{g_1,\ g_2\}$ nên

$$\boldsymbol{E}(f_t^u) = \frac{1}{2}g_1 + \frac{1}{2}g_2 = \frac{1}{2}f$$

$$\boldsymbol{E}(U_t) = \boldsymbol{E}(\frac{2}{t}\sum_{h=1}^{t} f_h^u) = \frac{2}{t}\sum_{h=1}^{t} \boldsymbol{E}(f_h^u) = \frac{2}{t}\sum_{h=1}^{t}\frac{1}{2} = \frac{2}{t}.\frac{t}{2}f = f$$

Vì vậy $U_t(\boldsymbol{\theta})$ là một ước lượng không chệch của $f(\boldsymbol{\theta})$. Với mỗi bước lặp t của OPE3, lấy f_t^u có phân phối đều từ $\{g_1,\ g_2\}$, tức là

$$P(f_t^u = g_1) = \frac{1}{2} \ ; \ P(f_t^u = g_2) = \frac{1}{2}$$

Thực hiện tương ứng giữa f_t^u và biến ngẫu nhiên X_t có phân phối đều từ $\{1, -1\}$:

$$P(X_t = 1) = \frac{1}{2} \ ; \ P(X_t = -1) = \frac{1}{2}$$

Sự tương ứng này là ánh xạ một-một. Vì vậy, $S_t = a_t - b_t$ có thể được biểu diễn dưới dạng $S_t = X_1 + \cdots + X_t$. Áp dụng luật logarit lặp LIL [103], ta có $S_t = \mathcal{O}(\sqrt{t \log t})$, dẫn đến $\frac{S_t}{t} \to 0$ khi $t \to +\infty$. Kết hợp với (2.4), ta có dãy $U_t \to f$ với xác suất 1, đồng thời từ (2.5), dãy đạo hàm $U_t' \to f'$ khi $t \to +\infty$. Sự hội tụ thu được cho mọi điểm $\boldsymbol{\theta} \in \overline{\Delta}_K$.

Xem xét

$$\begin{aligned}\langle U_t'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}\rangle &= \langle U_t'(\boldsymbol{\theta}_t) - f'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}\rangle + \langle f'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}\rangle \\ &= \frac{S_t}{t^2}\langle g_1^{'}(\boldsymbol{\theta}_t) - g_2^{'}(\boldsymbol{\theta}_t), \boldsymbol{e}_t^u - \boldsymbol{\theta}_t\rangle + \langle f'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}\rangle\end{aligned}$$

Ta có g_1 và g_2 là các hàm Lipschitz liên tục trên $\overline{\Delta}_K$. Vì vậy tồn tại một hằng số L sao cho:

$$\langle f'(z), y - z\rangle \leq f(y) - f(z) + L\|y - z\|^2, \ \forall\ y, z \in \overline{\Delta}_K$$

Do vậy xét:

$$\begin{aligned}\langle f'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}\rangle &= \langle f'(\boldsymbol{\theta}_t), \boldsymbol{\theta}_{t+1}^u - \boldsymbol{\theta}_t\rangle \\ &\leq f(\boldsymbol{\theta}_{t+1}^u) - f(\boldsymbol{\theta}_t) + L\|\boldsymbol{\theta}_{t+1}^u - \boldsymbol{\theta}_t\|^2 = f(\boldsymbol{\theta}_{t+1}^u) - f(\boldsymbol{\theta}_t) + L\|\frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}\|^2\end{aligned}$$

Ta có: $\boldsymbol{\theta}_{t+1} := \arg\max_{\boldsymbol{\theta} \in \{\boldsymbol{\theta}_{t+1}^u, \boldsymbol{\theta}_{t+1}^l\}} f(\boldsymbol{\theta})$, vì vậy

$$f(\boldsymbol{\theta}_{t+1}^u) \leq f(\boldsymbol{\theta}_{t+1})$$

Vì $\boldsymbol{e}_t^u$ và $\boldsymbol{\theta}_t$ thuộc Δ_K, nên $|\langle g_1'(\boldsymbol{\theta}_t) - g_2'(\boldsymbol{\theta}_t), \boldsymbol{e}_t^u - \boldsymbol{\theta}_t\rangle|$ và $\|\boldsymbol{e}_t^u - \boldsymbol{\theta}_t\|^2$ đều bị chặn trên với mọi t. Vì vậy tồn tại một hằng số $c_1 > 0$ sao cho

$$\langle U_t'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}\rangle \leq c_1\frac{|S_t|}{t^2} + f(\boldsymbol{\theta}_{t+1}) - f(\boldsymbol{\theta}_t) + \frac{c_1 L}{t^2} \tag{2.6}$$

Lấy tổng của (2.6) với mọi t, ta có

$$\sum_{t=1}^{+\infty}\frac{1}{t}\langle U_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t^u - \boldsymbol{\theta}_t\rangle \leq \sum_{t=1}^{+\infty} c_1\frac{|S_t|}{t^2} + f(\boldsymbol{\theta}_{+\infty}) - f(\boldsymbol{\theta}_1) + \sum_{t=1}^{+\infty}\frac{c_1 L}{t^2} \tag{2.7}$$

Bởi vì $f(\boldsymbol{\theta})$ bị chặn nên $f(\boldsymbol{\theta}_{+\infty})$ cũng bị chặn. Ghi nhớ rằng $S_t = \mathcal{O}(\sqrt{t\log t})$ [103], vì vậy $\sum_{t=1}^{+\infty} c_1\frac{|S_t|}{t^2}$ hội tụ với xác suất 1, và $\sum_{t=1}^{+\infty}\frac{L}{t^2}$ cũng bị chặn. Do đó, vế phải của (2.7) là xác định. Ngoài ra, $\langle U_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t^u\rangle > \langle U_t'(\boldsymbol{\theta}_t), \boldsymbol{\theta}_t\rangle$ với bất kỳ $t > 0$ bởi vì $e_t^u = \arg\max_{x \in \overline{\Delta}_K}\langle U_t'(\boldsymbol{\theta}_t), \boldsymbol{x}\rangle$. Do vậy, ta có:

$$0 \leq \sum_{t=1}^{+\infty}\frac{1}{t}\langle U_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t^u - \boldsymbol{\theta}_t\rangle < \infty \tag{2.8}$$

Nói cách khác, dãy $\sum_{t=1}^{+\infty}\frac{1}{t}\langle U_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t^u - \boldsymbol{\theta}_t\rangle$ hội tụ tới một hằng hữu hạn. Ta thấy $\langle U_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t^u - \boldsymbol{\theta}_t\rangle \geq 0$ với t bất kỳ. Nếu tồn tại một hằng số $c_2 > 0$ thỏa mãn $\langle U_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t^u - \boldsymbol{\theta}_t\rangle \geq c_2$ với t không xác định, khi đó dãy $\sum_{t=1}^{+\infty}\frac{1}{t}\langle U_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t^u - \boldsymbol{\theta}_t\rangle$ có thể không hội tụ đến hằng hữu hạn, điều này mâu thuẫn với (2.8). Vì vậy,

$$\langle U_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t^u - \boldsymbol{\theta}_t\rangle \to 0 \text{ as } t \to +\infty \tag{2.9}$$

Bởi vì $U'_t \to f'$ khi $t \to \infty$ và f' là liên tục, kết hợp với (2.9) ta có

$$\langle f'(\boldsymbol{\theta}_t), \boldsymbol{e}_t^u - \boldsymbol{\theta}_t \rangle \to 0 \text{ as } t \to +\infty \tag{2.10}$$

Sử dụng tiêu chuẩn "Frank-Wolfe gap" trong [72], từ (2.10) có $\boldsymbol{\theta}_t \to \boldsymbol{\theta}^*$ khi $t \to +\infty$. Nói cách khác, $\boldsymbol{\theta}_t$ hội tụ tới nghiệm tối ưu $\boldsymbol{\theta}^*$ của $f(\boldsymbol{\theta})$. □

Định lý 2.2 (Sự hội tụ của thuật toán OPE4)**.** *Xem xét hàm mục tiêu không lồi $f(\boldsymbol{\theta})$ của bài toán (2.1), cho trước văn bản $\boldsymbol{d}$, tham số $\boldsymbol{\beta}$ và α. Xét thuật toán OPE4, với xác suất 1, ta có:*

(i) Với $\boldsymbol{\theta} \in \Delta_K$, dãy hàm xấp xỉ $F_t(\boldsymbol{\theta})$ hội tụ tới $f(\boldsymbol{\theta})$ khi $t \to +\infty$,

(ii) Dãy nghiệm xấp xỉ $\boldsymbol{\theta}_t$ hội tụ tới điểm tối ưu cục bộ/điểm dừng của hàm $f(\boldsymbol{\theta})$.

Chứng minh. Ký hiệu

$$g_1(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}, \qquad g_2(\boldsymbol{\theta}) = (\alpha - 1) \sum_{k=1}^K \log \theta_k$$

Gọi a_t và b_t là số lần xuất hiện thành phần g_1 và g_2 sau t bước lặp để xây dựng dãy hàm xấp xỉ $\{U_t\}$.

Vì f_t^u được lựa chọn theo phân phối đều từ $\{g_1, g_2\}$ nên

$$E[f_t^u] = E[f_t^l] = \frac{1}{2}g_1 + \frac{1}{2}g_2 = \frac{1}{2}f$$
$$E[U_t] = E[\frac{2}{t}\sum_{h=1}^t f_h^u] = \frac{2}{t}\sum_{h=1}^t E[f_h^u] = \frac{2}{t}\sum_{h=1}^t \frac{1}{2}f = f$$

Tương tự, gọi c_t và d_t là số lần xuất hiện thành phần g_1 và g_2 sau t bước lặp để xây dựng dãy hàm xấp xỉ $\{L_t\}$. Vì f_t^l được lựa chọn theo phân phối đều từ $\{g_1, g_2\}$ nên:

$$E[f_t^u] = E[f_t^l] = \frac{1}{2}g_1 + \frac{1}{2}g_2 = \frac{1}{2}f$$
$$E[L_t] = E[\frac{2}{t}\sum_{h=1}^t f_h^l] = \frac{2}{t}\sum_{h=1}^t E[f_h^l] = \frac{2}{t}\sum_{h=1}^t \frac{1}{2}f = f$$

Do vậy

$$E[F_t] = \nu E[U_t] + (1-\nu)E[L_t] = \nu f + (1-\nu)f = f$$

Ký hiệu $S_t^u = a_t - b_t$, $S_t^l = c_t - d_t$ và $S_t = \max\{|S_t^u|,\ |S_t^l|\}$

Ta có

$$U_t = \frac{2}{t}(a_t g_1 + b_t g_2) \qquad a_t + b_t = t$$
$$L_t = \frac{2}{t}(c_t g_1 + d_t g_2) \qquad c_t + d_t = t$$
$$U_t - f = \frac{S_t^u}{t}(g_1 - g_2) \qquad L_t - f = \frac{S_t^l}{t}(g_1 - g_2)$$
$$U_t' - f' = \frac{S_t^u}{t}(g_1' - g_2') \qquad L_t' - f' = \frac{S_t^l}{t}(g_1' - g_2')$$

Do $F_t = \nu U_t + (1-\nu)L_t$ thu được:

$$\begin{aligned} F_t - f &= \nu(U_t - f) + (1-\nu)(L_t - f) \\ &= (\nu\frac{S_t^u}{t} + (1-\nu)\frac{S_t^l}{t})(g_1 - g_2) \\ F_t' - f' &= (\nu\frac{S_t^u}{t} + (1-\nu)\frac{S_t^l}{t})(g_1' - g_2') \end{aligned}$$

Vì vậy, F_t là một ước lượng không chệch của hàm mục tiêu đúng f. Áp dụng luật LIL [103] ta có $S_t^u = \mathcal{O}(\sqrt{t\log t})$ và $S_t^l = \mathcal{O}(\sqrt{t\log t})$, dẫn đến $\frac{S_t^u}{t} \to 0$ và $\frac{S_t^l}{t} \to 0$ khi $t \to +\infty$. Vì vậy, chúng tôi kết luận rằng dãy $U_t \to f$ và dãy đạo hàm $U_t' \to f$' khi $t \to +\infty$. Tương tự, xem xét với dãy $L_t \to f$, và dãy đạo hàm $L_t' \to f$' khi $t \to +\infty$.

Xem xét

$$\begin{aligned} &\langle F_t'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle = \langle F_t'(\boldsymbol{\theta}_t) - f'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle + \langle f'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle = \\ &= \langle (\nu\frac{S_t^u}{t} + (1-\nu)\frac{S_t^l}{t})(g_1'(\boldsymbol{\theta}_t) - g_2'(\boldsymbol{\theta}_t)), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle + \langle f'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle \end{aligned}$$

Chúng ta có g_1 và g_2 là hàm Lipschitz liên tục trên $\overline{\Delta}_K$. Vì vậy, tồn tại một hằng số L sao cho:

$$\langle f'(z), y - z\rangle \leq f(y) - f(z) + L\|y - z\|^2 \forall y, z \in \overline{\Delta}_K$$

Do đó:

$$\begin{aligned} \langle f'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle &= \langle f'(\boldsymbol{\theta}_t), \boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t\rangle \leq f(\boldsymbol{\theta}_{t+1}) - f(\boldsymbol{\theta}_t) + L\|\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t\|^2 \\ &= f(\boldsymbol{\theta}_{t+1}) - f(\boldsymbol{\theta}_t) + L\|\frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\|^2 \end{aligned}$$

Vì $\boldsymbol{e}_t$ và $\boldsymbol{\theta}_t$ đều thuộc $\overline{\Delta}_K$ nên $\langle g_1'(\boldsymbol{\theta}_t) - g_2'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t\rangle$ và $\|\boldsymbol{e}_t^u - \boldsymbol{\theta}_t\|^2$ bị chặn. Do đó, tồn tại một hằng $c_1 > 0$ sao cho:

$$\langle F_t'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle \leq c_1\frac{S_t}{t^2} + f(\boldsymbol{\theta}_{t+1}) - f(\boldsymbol{\theta}_t) + \frac{c_1 L}{t^2} \tag{2.11}$$

Lấy tổng hai vế của (2.11) với mọi t, ta có

$$\sum_{t=1}^{+\infty} \frac{1}{t}\langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t\rangle \leq \sum_{t=1}^{+\infty} c_1 \frac{S_t}{t^2} + f(\boldsymbol{\theta}^*) - f(\boldsymbol{\theta}_1) + \sum_{t=1}^{+\infty} \frac{c_1 L}{t^2} \tag{2.12}$$

Bởi vì $f(\boldsymbol{\theta})$ bị chặn nên $f(\boldsymbol{\theta}^*)$ bị chặn. Chúng ta thấy $S_t = \mathcal{O}(\sqrt{t \log t})$ theo tài liệu [103], vì vậy dãy $\sum_{t=1}^{+\infty} c_1 \frac{S_t}{t^2}$ hội tụ với xác suất 1 và tổng $\sum_{t=1}^{+\infty} \frac{L}{t^2}$ cũng bị chặn. Vì vậy, vế phải của (2.12) là hữu hạn.

Bởi vì $e_t = \arg\max_{x \in \overline{\Delta}_K} \langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{x}\rangle$ nên $\langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t\rangle > \langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{\theta}_t\rangle$ với bất kỳ $t > 0$. Vì vậy thu được:

$$0 \leq \sum_{t=1}^{+\infty} \frac{1}{t}\langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t\rangle < +\infty \tag{2.13}$$

Nói cách khác, dãy $\sum_{t=1}^{+\infty} \frac{1}{t}\langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t\rangle$ hội tụ tới một hằng số hữu hạn.

Ta thấy $\langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t\rangle \geq 0$ với bất kỳ t. Nếu tồn tại một hằng số $c_3 > 0$ thỏa mãn $\langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t\rangle \geq c_3$ với t nhận giá trị vô hạn, khi đó chuỗi $\sum_{t=1}^{+\infty} \frac{1}{t}\langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t\rangle$ không thể hội tụ tới một hằng hữu hạn, điều này trái ngược với (2.13). Vì vậy

$$\langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t\rangle \to 0 \text{ khi } t \to +\infty \tag{2.14}$$

Bởi vì $F_t' \to f'$ khi $t \to \infty$ và f' là các hàm liên tục, kết hợp với (2.14) có:

$$\langle f'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t\rangle \to 0 \text{ khi } t \to +\infty.$$

Sử dụng tiêu chuẩn "Frank-Wolfe gap" [72], ta có $\boldsymbol{\theta}_t \to \boldsymbol{\theta}^*$ as $t \to +\infty$. Như vậy, $\boldsymbol{\theta}_t$ hội tụ theo xác suất đến điểm dừng/cực đại địa phương $\boldsymbol{\theta}^*$ of hàm mục tiêu $f(\boldsymbol{\theta})$. □

2.6. Mở rộng thuật toán đề xuất cho bài toán tối ưu không lồi

Phân tích đặc điểm của OPE3 và OPE4, chúng tôi nhận thấy có thể sửa đổi OPE3 và OPE4 dễ dàng để giải bài toán tối ưu không lồi tổng quát có dạng như trong (2.2), tức là giải bài toán tối đa hóa hàm mục tiêu có dạng $f(x) = g_1(x) + g_2(x)$ trên miền ràng buộc Ω. Do đó bước tìm e_t trong OPE3 hay OPE4 sẽ là một bài toán quy hoạch tuyến tính có thể giải được. Xét bài toán tối ưu không lồi tổng quát:

$$\max_{x \in \Omega} [f(x) = g_1(x) + g_2(x)] \tag{2.15}$$

Chi tiết của thuật toán OPE3 và OPE4 tổng quát để giải bài toán (2.15) được trình bày trong Thuật toán 2.7 và Thuật toán 2.8.

Thuật toán 2.7 OPE3 giải bài toán tối ưu không lồi tổng quát

Đầu ra: $\boldsymbol{x}^*$ là nghiệm cực đại hóa của hàm $f(\boldsymbol{x}) = g_1(\boldsymbol{x}) + g_2(\boldsymbol{x})$ trên miền Ω

Khởi tạo $\boldsymbol{x}_1$ thuộc Ω
$f_1^l := g_1(\boldsymbol{x})$; $f_1^u := g_2(\boldsymbol{x})$
for $t = 2, 3, .., \infty$ **do**
  Lấy f_t^u có phân phối đều từ $\{g_1(\boldsymbol{x})\ ;\ g_2(\boldsymbol{x})\}$
  $U_t := \frac{2}{t}\sum_{h=1}^{t} f_h^u$
  $\boldsymbol{a}_t^u := \arg\max_{\boldsymbol{x}\in\Omega}\langle U_t^{'}(\boldsymbol{x}_t), \boldsymbol{x}\rangle$
  $\boldsymbol{x}_{t+1}^u := \boldsymbol{x}_t + \frac{\boldsymbol{a}_t^u - \boldsymbol{x}_t}{t}$
  Lấy f_t^l có phân phối đều từ $\{g_1(\boldsymbol{x})\ ;\ g_2(\boldsymbol{x})\}$
  $L_t := \frac{2}{t}\sum_{h=1}^{t} f_h^l$
  $\boldsymbol{a}_t^l := \arg\max_{\boldsymbol{x}\in\Omega}\langle L_t^{'}(\boldsymbol{x}_t), \boldsymbol{x}\rangle$
  $\boldsymbol{x}_{t+1}^l := \boldsymbol{x}_t + \frac{\boldsymbol{a}_t^l - \boldsymbol{x}_t}{t}$
  $\boldsymbol{x}_{t+1} := \arg\max_{\boldsymbol{x}\in\{\boldsymbol{x}_{t+1}^u, \boldsymbol{x}_{t+1}^l\}} f(\boldsymbol{x})$
end for

Thuật toán 2.8 OPE4 giải bài toán tối ưu không lồi tổng quát

Đầu vào: Tham số tổ hợp $\nu \in (0, 1)$
Đầu ra: $\boldsymbol{x}^*$ là nghiệm cực đại hóa của hàm $f(\boldsymbol{x}) = g_1(\boldsymbol{x}) + g_2(\boldsymbol{x})$ trên miền Ω

Khởi tạo $\boldsymbol{x}_1$ thuộc Ω
$f_1^l := g_1(\boldsymbol{x})$; $f_1^u := g_2(\boldsymbol{x})$
for $t = 2, 3, .., \infty$ **do**
  Lấy f_t^u có phân phối đều từ $\{g_1(\boldsymbol{x})\ ;\ g_2(\boldsymbol{x})\}$
  $U_t := \frac{2}{t}\sum_{h=1}^{t} f_h^u$
  Lấy f_t^l có phân phối đều từ $\{g_1(\boldsymbol{x})\ ;\ g_2(\boldsymbol{x})\}$
  $L_t := \frac{2}{t}\sum_{h=1}^{t} f_h^l$
  Lấy $F_t := \nu U_t + (1-\nu)L_t$
  $\boldsymbol{a}_t := \arg\max_{\boldsymbol{x}\in\Omega}\langle F_t^{'}(\boldsymbol{x}_t), \boldsymbol{x}\rangle$
  $\boldsymbol{x}_{t+1} := \boldsymbol{x}_t + \frac{\boldsymbol{a}_t - \boldsymbol{x}_t}{t}$
end for

Sự hội tụ của OPE3 và OPE4 trong trường hợp tổng quát này có thể chứng minh tương tự như Định lý 2.1 và Định lý 2.2 bởi các quá trình chứng minh không bị phụ thuộc vào hàm thành phần $g_1(\boldsymbol{x})$ và $g_2(\boldsymbol{x})$ cụ thể.

2.7. Kết luận chương 2

Chúng tôi tổng kết một số kết quả đạt được của chương như sau:

- Trong chương này chúng tôi đã đề xuất bốn thuật toán tối ưu mới OPE1, OPE2, OPE3 và OPE4 để giải bài toán suy diễn hậu nghiệm với mô hình chủ đề ẩn LDA, trong đó OPE3 và OPE4 thường hiệu quả hơn thuật toán OPE. Do vậy, OPE3 và OPE4 đã được chúng tôi nghiên cứu một cách nghiêm túc và đầy đủ trên hai mặt lý thuyết và thực nghiệm.
- Các cải tiến khai thác theo hướng tiếp cận ngẫu nhiên hóa thông qua việc xem xét hàm mục tiêu là các xấp xỉ ngẫu nhiên, sử dụng phân phối đều

phù hợp với xu thế tiếp cận phương pháp ngẫu nhiên giải bài toán MAP không lồi;

- Hơn nữa, OPE3 và OPE4 hoàn toàn có thể mở rộng dễ dàng để giải bài toán quy hoạch DC [104], một lớp bài toán tối ưu không lồi khó giải

$$\min_{x\in\Omega}[f(x) = g(x) - h(x)]$$

bằng cách đặt tương ứng $g_1 := g$ và $g_2 := -h$.

Các kết quả trình bày trong chương 2 được chúng tôi trình bày trong bài báo "Stochastic bounds for inference in topic models" xuất bản trong kỷ yếu hội thảo quốc tế ICTA năm 2016 và bài báo "Some methods for posterior inference in topic models" xuất bản trên tạp chí RD-ICT của Bộ thông tin truyền thông năm 2018.

Chương 3

TỔNG QUÁT HÓA THUẬT TOÁN TỐI ƯU GIẢI BÀI TOÁN MAP KHÔNG LỒI TRONG MÔ HÌNH CHỦ ĐỀ

Trong chương này, nghiên cứu sinh tiếp tục đề xuất thuật toán GOPE theo hướng ngẫu nhiên thông qua sử dụng phân phối Bernoulli hợp lý và xấp xỉ ngẫu nhiên để giải bài toán MAP không lồi. Sự hiệu quả của thuật toán GOPE được xem xét trên cả hai mặt lý thuyết và thực nghiệm, trong đó sử dụng GOPE là thuật toán suy diễn cho bài toán MAP trong các mô hình chủ đề.

3.1. Giới thiệu

Xem xét bài toán ước lượng MAP trong các mô hình đồ thị xác suất:

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} \left[\log P(D|\mathbf{x}) + \log P(\mathbf{x})\right] \tag{3.1}$$

Ký hiệu $g_1(\mathbf{x}) := \log P(D|\mathbf{x})$ và $g_2(\mathbf{x}) := \log P(\mathbf{x})$, bài toán (3.1) được đưa về bài toán tối ưu có dạng:

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} \left[f(\mathbf{x}) = g_1(\mathbf{x}) + g_2(\mathbf{x})\right] \tag{3.2}$$

trong đó hàm mục tiêu $f(\mathbf{x})$ được phân tích thành tổng của hai thành phần $g_1(\mathbf{x})$ và $g_2(\mathbf{x})$. Bài toán (3.2) là khó giải khi hàm mục tiêu $f(\mathbf{x})$ là hàm không lõm.

Một ví dụ điển hình cho bài toán (3.2) chính là bài toán MAP trong mô hình chủ đề LDA:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Delta_K} \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^{K} \log \theta_k \tag{3.3}$$

trong đó α là tham số của phân phối tiên nghiệm Dirichlet. [37] đã chỉ ra rằng bài toán (3.3) thuộc lớp NP-khó khi tham số $\alpha < 1$. Trong trường hợp $\alpha \geq 1$, dễ dàng chỉ ra rằng bài toán (3.3) là tối ưu lõm, khi đó có thể được giải quyết trong thời gian đa thức. Thật không may, trong thực tế mô hình LDA, tham số α thường nhỏ, tức $\alpha < 1$ [42, 92], khiến cho bài toán (3.3) trở thành bài toán tối ưu không lõm (non-concave). Đó là lý do tại sao (3.3) không khả thi trong các trường hợp xấu.

Chương 2 đã xem xét bài toán (3.3) với các cải tiến OPE1, OPE2, OPE3 và OPE4, đặc biệt OPE3 và OPE4 là hai thuật toán hiệu quả nhất. Chúng tôi tiếp tục tiếp cận theo hướng ngẫu nhiên hóa để đề xuất các thuật toán hiệu quả giải bài toán MAP không lồi. Đồng thời đảm bảo thuật toán đề xuất có tính linh hoạt, dễ dàng mở rộng cho bài toán không lồi khác xuất hiện trong học máy. Chúng tôi nhận thấy phân phối Bernoulli là phân phối rời rạc đơn giản nhưng tổng quát hơn phân phối đều và có nhiều ứng dụng trong thực tế. Đây là một ý tưởng để chúng tôi cải tiến thuật toán OPE và đưa ra thuật toán mới đảm bảo tính tổng quát và hiệu quả hơn dựa trên phân phối Bernoulli và xấp xỉ ngẫu nhiên.

3.2. Thuật toán Generalized Online Maximum a Posteriori Estimation

Xét bài toán MAP (3.3) với mô hình chủ đề:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Delta_K} \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^K \log \theta_k$$

Chúng tôi giới thiệu thuật toán mới đặt tên là GOPE (viết tắt của Generalized Online Maximum a Posteriori Estimation) để giải bài toán MAP (3.3).

Ký hiệu

$$g_1(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} \ , \ g_2(\boldsymbol{\theta}) = (\alpha - 1) \sum_{k=1}^K \log \theta_k$$

Khi đó

$$f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^K \log \theta_k = g_1(\boldsymbol{\theta}) + g_2(\boldsymbol{\theta})$$

với $g_1(\boldsymbol{\theta})$ và $g_2(\boldsymbol{\theta})$ tương ứng là thành phần likelihood và prior.

Như đã biết, OPE hoạt động bằng cách lựa chọn ngẫu nhiên thành phần likelihood $g_1(\boldsymbol{\theta})$ hay prior $g_2(\boldsymbol{\theta})$ tại mỗi bước lặp t theo phân phối đều (xác suất lựa chọn thành phần $g_1(\boldsymbol{\theta})$ và $g_2(\boldsymbol{\theta})$ là như nhau và bằng $1/2$):

$$f_t := \text{uniform}\{g_1, g_2\} \ , \forall t = 1, 2, ..$$

sau đó xây dựng dãy hàm xấp xỉ $F_t(\boldsymbol{\theta})$ của hàm mục tiêu đúng $f(\boldsymbol{\theta})$ theo công thức:

$$F_t := \frac{2}{t} \sum_{k=1}^t f_k$$

sao cho đảm bảo $F_t \to f$ khi $t \to \infty$. Ở đây thành phần likelihood $g_1(\boldsymbol{\theta})$ đại diện cho tri thức ta quan sát được về đối tượng là văn bản, thành phần tiên nghiệm (prior) $g_2(\boldsymbol{\theta})$ đại diện cho tri thức ta biết trước về văn bản. Với cách thực hiện của OPE, về mặt trung bình, F_t có tỉ lệ likelihood và prior bằng nhau. Tuy nhiên, trong thực tế, khi suy diễn về một đối tượng nào đó, con người có thể dùng nhiều hơn thành phần likelihood nếu ta quan sát được nhiều thông tin về đối tượng, hoặc dùng nhiều hơn thành phần prior nếu ta biết ít thông tin về đối tượng. Đây là một suy diễn rất tự nhiên. Để triển khai ý tưởng đó, chúng ta có thể sử dụng phân phối Bernoulli với xác suất p để hiệu chỉnh tỉ lệ của thành phần prior và likelihood thay vì sử dụng phân phối đều như trong OPE. Tức là, chúng ta ấn định xác suất để chọn được thành phần g_1 là p và xác suất chọn được thành phần g_2 là $1-p$ với $p \in (0,1)$, biết rằng phân phối Bernoulli tổng quát hơn phân phối đều.

Mục tiêu cải tiến của chúng tôi là đề xuất thuật toán mới không những giữ lại những đặc tính tốt của OPE mà hiệu quả hơn OPE. Vì vậy, chúng tôi có ý tưởng thay thế phân phối đều trong lựa chọn mẫu bằng phân phối Bernoulli với xác suất $p \in (0,1)$ thích hợp sao cho hàm xấp xỉ $F_t(\boldsymbol{\theta})$ vẫn hội tụ về hàm mục tiêu $f(\boldsymbol{\theta})$.

Giả sử cho trước xác suất Bernoulli $p \in (0,1)$. Trước hết, để đảm bảo hàm xấp xỉ $F_t(\boldsymbol{\theta}) \to f(\boldsymbol{\theta})$ khi $t \to \infty$, tiến hành hiệu chỉnh likelihood $g_1(\boldsymbol{\theta})$ và prior $g_2(\boldsymbol{\theta})$ ban đầu như sau:

$$G_1(\boldsymbol{\theta}) := \frac{g_1(\boldsymbol{\theta})}{p} \ ; \ G_2(\boldsymbol{\theta}) := \frac{g_2(\boldsymbol{\theta})}{1-p}$$

Khi đó $G_1(\boldsymbol{\theta})$ và $G_2(\boldsymbol{\theta})$ tương ứng là likelihood và prior hiệu chỉnh theo tham số p của phân phối Bernoulli. Tham số p được sử dụng để hiệu chỉnh tỉ lệ của likelihood và prior trong thuật toán suy diễn. GOPE được trình bày chi tiết trong Thuật toán 3.1.

Ký hiệu T là số bước lặp tối thiểu thực hiện thuật toán GOPE. Bởi vì GOPE là thuật toán ngẫu nhiên nên về mặt lý thuyết luôn xem xét trong ngữ cảnh $T \to \infty$. Tại bước lặp thứ t trong Thuật toán 3.1, chúng tôi lấy $f_t(\boldsymbol{\theta})$ tuân theo phân phối Bernoulli của hai thành phần likelihood và prior đã được hiệu chỉnh và $F_t(\boldsymbol{\theta})$ là hàm xấp xỉ cần xây dựng. Với mỗi bước lặp $t, (t = 1, 2, .., T)$, chúng tôi lấy f_t có phân phối Bernoulli với xác suất $p \in (0,1)$ từ $\{G_1 \ , \ G_2\}$. Ta thấy T lần lặp tương ứng chính là thực hiện T phép thử Bernoulli với xác suất $p \in (0,1)$:

$$P(f_t = G_1) = p \ , \ P(f_t = G_2) = 1 - p \ , \ \forall t = 1, 2, .., T$$

Thuật toán 3.1 GOPE: Generalized Online maximum a Posteriori Estimation

Đầu vào: Văn bản $\mathbf{d}$, tham số mô hình $\{\boldsymbol{\beta}, \alpha\}$ và tham số Bernoulli $p \in (0,1)$
Đầu ra: $\boldsymbol{\theta}^*$ là điểm cực đại của hàm $f(\boldsymbol{\theta}) = g_1(\boldsymbol{\theta}) + g_2(\boldsymbol{\theta})$

Khởi tạo $\boldsymbol{\theta}_1$ trong miền $\overline{\Delta}_K$
$G_1 := \frac{g_1}{p}$; $G_2 := \frac{g_2}{1-p}$
for $t = 1, 2, \ldots, T$ **do**
    Lấy f_t có phân phối Bernoulli từ $\{G_1, G_2\}$ với xác suất p trong đó

$$\{P(f_t = G_1) = p; P(f_t = G_2) = 1 - p\}$$

    $F_t(\boldsymbol{\theta}) := \frac{1}{t}\sum_{h=1}^{t} f_h$
    $\boldsymbol{e}_t := \arg\max_{\boldsymbol{x} \in \overline{\Delta}_K} \langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{x} \rangle$
    $\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}$
end for

Theo lý thuyết thống kê, hiệu quả đạt được tốt hơn khi số phép thử Bernoulli (tức số lần lặp) T tăng (ít nhất là 20) và tham số p lựa chọn không quá gần 0 hoặc 1.

Giả sử tiến hành thực hiện thuật toán GOPE với ít nhất T bước lặp. Xét tại bước lặp thứ t, $(t = 1, 2, .., T)$:

1. Thực hiện t phép thử Bernoulli xác suất $p \in (0,1)$:

$$\{P(f_h = G_1) = p \ , \ P(f_h = G_2) = 1 - p\} \ \forall h = 1, ..., t$$

   Ta có: $E[f_h] = f \ , \ \forall h = 1, .., t.$

2. Xây dựng một dãy hàm xấp xỉ ngẫu nhiên:

$$F_t := \frac{1}{t}\sum_{h=1}^{t} f_h$$

   Ta có: $E[F_t] = E[\frac{1}{t}\sum_{h=1}^{t} f_h] = f$. Như vậy, F_t là trung bình mẫu của các hàm mẫu $\{f_1, f_2, .., f_t\}$ và F_t đảm bảo hội tụ đến hàm mục tiêu f khi $t \to \infty$. Điều này được chỉ rõ trong phần chứng minh Định lý 3.1.

Tham số p đóng vai trò điều chỉnh tỷ lệ của thành phần likelihood và prior đóng góp vào hàm xấp xỉ F_t. Nếu p lớn thì khả năng lựa chọn được thành phần likelihood G_1 nhiều hơn G_2 trong xây dựng F_t. Ngược lại, nếu p nhỏ thì khả năng sẽ lựa chọn được thành phần prior G_2 nhiều hơn G_1 trong xây dựng hàm F_t. Chúng ta sử dụng đặc điểm này để lựa chọn p phù hợp nhất cho thuật toán GOPE trong mỗi bài toán áp dụng. Ta thấy OPE chính là một trường hợp đặc biệt của GOPE khi tham số p được lựa chọn là 0.5, nên OPE không có tính linh hoạt khi áp dụng cho nhiều bài toán và nhiều bộ dữ liệu khác nhau. GOPE có khả năng thích nghi tốt với các bộ dữ liệu khác nhau thông qua việc điều chỉnh

Thuật toán 3.2 Online-GOPE học mô hình LDA từ dữ liệu lớn

Đầu vào: Tập dữ liệu huấn luyện $\mathcal{C}$ với D văn bản, $K, \alpha, \eta, \tau > 0, \kappa \in (0.5, 1]$
Đầu ra: $\boldsymbol{\lambda}$

1: Khởi tạo $\boldsymbol{\lambda}^{\mathbf{0}}$ ngẫu nhiên
2: **for** $t = 1, 2, \ldots, \infty$ **do**
3: Lấy mẫu nhỏ $\mathcal{C}_t$ gồm S văn bản.
4: Sử dụng BOPE để suy diễn hậu nghiệm cho mỗi văn bản $\mathbf{d} \in \mathcal{C}_t$, cho bởi biến toàn cục $\boldsymbol{\beta}^{t-1} \propto \boldsymbol{\lambda}^{t-1}$ trong bước trước đó để nhận được chủ đề hỗn hợp $\boldsymbol{\theta}_d$. Tính toán ϕ^d theo

$$\phi_{dkj} \propto \theta_{dk}\beta_{kj}$$

5: Với mỗi $k \in \{1, 2, .., K\}$, biến toàn cục $\hat{\boldsymbol{\lambda}}_k$ cho $\mathcal{C}_t$ bởi

$$\hat{\lambda}_{kj} = \eta + \frac{D}{S}\sum_{d \in \mathcal{C}_t} d_j \phi_{djk}$$

6: Cập nhật biến toàn cục

$$\boldsymbol{\lambda}^t := (1 - \rho_t)\boldsymbol{\lambda}^{t-1} + \rho_t \hat{\boldsymbol{\lambda}}$$

trong đó $\rho_t = (t + \tau)^{-\kappa}$
7: **end for**

giá trị của tham số $p \in (0, 1)$. Điều này sẽ được thể hiện rõ hơn trong phần thực nghiệm. Chúng tôi sẽ chỉ ra rằng GOPE duy trì lợi thế chính của OPE là sự đảm bảo về chất lượng và tốc độ hội tụ, hơn nữa đem đến một sự tổng quát cao hơn OPE nhờ đóng góp của phân phối Bernoulli. Các đặc tính tốt này không được xác định đối với các phương pháp hiện có khi áp dụng cho bài toán suy diễn hậu nghiệm trong các mô hình chủ đề.

GOPE đóng vai trò là bước suy diễn cốt lõi khi học mô hình LDA. Chúng tôi sử dụng GOPE thay cho OPE trong thuật toán học Online-OPE [28] và nhận được thuật toán học ngẫu nhiên mới đặt tên là Online-GOPE. Online-GOPE được trình bày chi tiết trong Thuật toán 3.2.

3.3. Sự hội tụ của thuật toán GOPE

Trong phần này, chúng tôi đưa ra bằng chứng về sự hội tụ của GOPE. Đây cũng là một điểm mạnh của đề xuất bởi rất nhiều thuật toán đề xuất trước đây mới dừng lại ở minh chứng sự hiệu quả về thực nghiệm mà chưa đưa ra các đảm bảo về mặt lý thuyết.

Định lý 3.1 (Sự hội tụ của thuật toán GOPE)**.** *Xét hàm mục tiêu $f(\boldsymbol{\theta})$ trong bài toán* (3.3)*, cho trước văn bản $\boldsymbol{d}$, tham số mô hình $\{\boldsymbol{\beta}, \alpha\}$ và tham số Bernoulli $p \in (0, 1)$. Xét GOPE, với xác suất 1, ta có:*

(i) Với bất kỳ $\boldsymbol{\theta} \in \Delta_K$, dãy hàm $F_t(\boldsymbol{\theta})$ hội tụ tới $f(\boldsymbol{\theta})$ khi $t \to +\infty$;

(ii) Dãy nghiệm xấp xỉ $\boldsymbol{\theta}_t$ hội tụ tới điểm dừng/cực đại địa phương của hàm mục tiêu $f(\boldsymbol{\theta})$ với tốc độ hội tụ là $\mathcal{O}(1/t)$.

Chứng minh. Việc chứng minh Định lý 3.1 được dựa vào [28] và Định lý 2.1 và Định lý 2.2 trong Chương 2. Chúng tôi chứng minh rằng GOPE cũng hội tụ tới điểm dừng/cực đại địa phương của hàm mục tiêu.

Nhắc lại một số ký hiệu: $B(t,p)$ ký hiệu cho phân phối nhị thức với tham số t và p, $N(\mu,\ \sigma^2)$ ký hiệu cho phân phối chuẩn với tham số μ và σ, $E(X)$ là kỳ vọng của biến ngẫu nhiên X, $D(X)$ là phương sai của biến ngẫu nhiên X.

Theo tìm hiểu, với các bài toán tối ưu không lồi không ràng buộc, véc tơ đạo hàm của hàm mục tiêu $\|\nabla f(\boldsymbol{\theta})\|$ được dùng là tiêu chuẩn để đánh giá sự hội tụ, bởi vì $\|\nabla f(\boldsymbol{\theta})\| \to 0$ đưa đến sự hội tụ đến một điểm dừng. Tuy nhiên, tiêu chuẩn này không thể được sử dụng cho các bài toán tối ưu không lồi có ràng buộc. Thay vào đó, chúng tôi sử dụng tiêu chuẩn "Frank-Wolfe gap" đề cập trong [72]. Nhắc lại

$$G_1(\boldsymbol{\theta}) := g_1(\boldsymbol{\theta})/p\ ;\ G_2(\boldsymbol{\theta}) := g_2(\boldsymbol{\theta})/1-p$$

là likelihood và prior hiệu chỉnh tương ứng. Nhận thấy:

$$f(\boldsymbol{\theta}) = g_1(\boldsymbol{\theta}) + g_2(\boldsymbol{\theta}) = pG_1(\boldsymbol{\theta}) + (1-p)G_2(\boldsymbol{\theta})$$

Tiến hành thực hiện t phép thử Bernoulli, trong đó tại phép thử thứ $h, (h = 1, .., t)$ lấy f_h có phân phối Bernoulli từ $\{G_1, G_2\}$ trong đó:

$$P(f_h = G_1) = p\ ;\ P(f_h = G_2) = 1-p$$

Gọi a_t là số lần lấy được G_1 sau t bước lặp. Khi đó $b_t = t - a_t$ là số lần lấy được G_2 sau t bước lặp. Khi đó a_t có phân phối nhị thức với tham số t và p, tức $a_t \sim B(t,p)$ và có các đặc trưng:

$$E[a_t] = tp\ ,\ D[a_t] = tp(1-p)$$

Đặt $S_t = a_t - tp$. Theo tính chất của phân phối nhị thức và định lý Moivre-Laplace [105], ta có $S_t \to N(0, tp(1-p))$ khi $t \to \infty$. Hơn nữa, với xác suất 1, dãy $\frac{S_t}{t} \to 0$ khi $t \to \infty$. Ta có

$$\begin{aligned} F_t &= \frac{1}{t}(a_t G_{_1} + (t-a_t)G_2) \\ F_t - f &= \frac{S_t}{t}(G_1 - G_2) \\ F_t' - f' &= \frac{S_t}{t}(G_1' - G_2') \end{aligned} \tag{3.4}$$

Từ (3.4) thu được $F_t \to f$ khi $t \to +\infty$ với xác suất 1. Theo GOPE có công thức nghiệm xấp xỉ được tính theo lược đồ lặp:

$$\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}$$

tuân theo lược đồ lặp: $\boldsymbol{\theta}_{t+1} = \phi(\boldsymbol{\theta}_t)$ trong đó hàm $\phi(\boldsymbol{\theta}_t) = \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}$. Theo nguyên lý ánh xạ co Banach, ta có dãy nghiệm $\{\boldsymbol{\theta}_t\}$ hội tụ về điểm bất động $\boldsymbol{\theta}^*$ nào đó. Đồng thời có đánh giá:

$$\|\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t\| = \|\frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\| = \frac{\|\boldsymbol{e}_t - \boldsymbol{\theta}_t\|}{t} = \mathcal{O}(\frac{1}{t})$$

Như vậy $\boldsymbol{\theta}_t \to \boldsymbol{\theta}^*$ với tốc độ hội tụ là $\mathcal{O}(1/t)$.

Xét:

$$\begin{aligned}\langle F_t'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle &= \langle F_t'(\boldsymbol{\theta}_t) - f'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle + \langle f'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle \\ &= \frac{S_t}{t^2}\langle G_1'(\boldsymbol{\theta}_t) - G_2'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t\rangle + \langle f'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle\end{aligned}$$

Chú ý rằng $g_1(\boldsymbol{\theta})$ và $g_2(\boldsymbol{\theta})$ là các hàm Lipschitz liên tục trên $\overline{\Delta}_K$. Vì vậy, tồn tại hằng số L sao cho:

$$\langle f'(z), y - z\rangle \leq f(y) - f(z) + L\|y - z\|^2 \forall y, z \in \overline{\Delta}_K$$

. Ta có

$$\begin{aligned}\langle f'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle = \langle f'(\boldsymbol{\theta}_t), \boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t\rangle \leq f(\boldsymbol{\theta}_{t+1}) - f(\boldsymbol{\theta}_t) + L\|\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t\|^2 = \\ = f(\boldsymbol{\theta}_{t+1}) - f(\boldsymbol{\theta}_t) + \frac{L}{t^2}\|\boldsymbol{e}_t - \boldsymbol{\theta}_t\|^2\end{aligned}$$

Vì $\boldsymbol{e}_t$ và $\boldsymbol{\theta}_t$ đều thuộc về $\overline{\Delta}_K$, nên $|\langle G_1'(\boldsymbol{\theta}_t) - G_2'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t\rangle|$ và $\|\boldsymbol{e}_t - \boldsymbol{\theta}_t\|^2$ bị chặn trên với t bất kỳ. Vì vậy, tồn tại một hằng số $c_1 > 0$ sao cho

$$\langle F_t'(\boldsymbol{\theta}_t), \frac{\boldsymbol{e}_t - \boldsymbol{\theta}_t}{t}\rangle \leq c_1\frac{|S_t|}{t^2} + f(\boldsymbol{\theta}_{t+1}) - f(\boldsymbol{\theta}_t) + \frac{c_1 L}{t^2} \tag{3.5}$$

Lấy tổng hai vế của (3.5) với mọi t, ta có:

$$\sum_{h=1}^{t}\frac{1}{h}\langle F_h'(\boldsymbol{\theta}_h), \boldsymbol{e}_h - \boldsymbol{\theta}_h\rangle \leq \sum_{h=1}^{t} c_1\frac{|S_h|}{h^2} + f(\boldsymbol{\theta}_{t+1}) - f(\boldsymbol{\theta}_1) + \sum_{h=1}^{t}\frac{c_1 L}{h^2} \tag{3.6}$$

Khi $t \to +\infty$, $f(\boldsymbol{\theta}_t) \to f(\boldsymbol{\theta}^*)$ do sự liên tục của hàm $f(\boldsymbol{\theta})$. Kết quả (3.6) có nghĩa là

$$\sum_{h=1}^{+\infty}\frac{1}{h}\langle F_h'(\boldsymbol{\theta}_h), \boldsymbol{e}_h - \boldsymbol{\theta}_h\rangle \leq \sum_{h=1}^{+\infty} c_1\frac{|S_h|}{h^2} + f(\boldsymbol{\theta}^*) - f(\boldsymbol{\theta}_1) + \sum_{h=1}^{+\infty}\frac{c_1 L}{h^2} \tag{3.7}$$

Sử dụng [28] và theo luật số lớn, ta có $\sum_{h=1}^{\infty} c_1 \frac{|S_h|}{h^2}$ hội tụ theo nghĩa xác suất. Ngoài ra, số hạng $\sum_{h=1}^{+\infty} \frac{1}{h^2}$ là bị chặn trên. Vì vậy, $\sum_{h=1}^{+\infty} \frac{1}{h} \langle F_h'(\boldsymbol{\theta}_h), \boldsymbol{e}_h - \boldsymbol{\theta}_h \rangle$ cũng bị chặn trên.

Bởi vì $\boldsymbol{e}_t = \arg\max_{\boldsymbol{x} \in \Delta_K} \langle F_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} \rangle$, nên $\langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t \rangle \geq 0$.

Nếu tồn tại $t_0 > 0$, $c_3 > 0$ sao cho $\langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t \rangle \geq c_3 \forall t > t_0$, khi đó $\sum_{t=1}^{\infty} \frac{1}{t} \langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t \rangle > \sum_{t=1}^{\infty} \frac{c_3}{t}$. Đồng thời vì $\sum_{t=1}^{\infty} \frac{1}{t}$ không bị chặn trên, nên $\sum_{h=1}^{+\infty} \frac{1}{h} \langle F_h'(\boldsymbol{\theta}_h), \boldsymbol{e}_h - \boldsymbol{\theta}_h \rangle \to \infty$, điều này mâu thuẫn với kết luận ở trên.

Do đó, $\langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t \rangle \to 0$ khi $t \to \infty$. Mặt khác,

$$\begin{aligned}
\langle F_t'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t \rangle &= \langle f'(\boldsymbol{\theta}_t) + \frac{S_t}{t}(G_1'(\boldsymbol{\theta}_t) - G_2'(\boldsymbol{\theta}_t)), \boldsymbol{e}_t - \boldsymbol{\theta}_t \rangle \\
&= \langle f'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t \rangle + \langle \frac{S_t}{t}(G_1'(\boldsymbol{\theta}_t) - G_2'(\boldsymbol{\theta}_t)), \boldsymbol{e}_t - \boldsymbol{\theta}_t \rangle
\end{aligned}$$

Bởi vì $\frac{S_t}{t} \to 0$ nên $\langle f'(\boldsymbol{\theta}_t), \boldsymbol{e}_t - \boldsymbol{\theta}_t \rangle \to 0$. Áp dụng tiêu chuẩn "Frank-Wolfe gap" ta thu được $\boldsymbol{\theta}^*$ là điểm dừng hoặc là điểm cực đại địa phương của hàm mục tiêu $f(\boldsymbol{\theta})$. □

3.4. Đánh giá thực nghiệm

Trong phần này, chúng tôi sẽ điều tra hiệu quả của GOPE về mặt thực nghiệm khi giải bài toán hậu nghiệm với mô hình chủ đề LDA với các bộ dữ liệu lớn thu thập từ thế giới thực. Điều tra hiệu quả của GOPE thông qua hiệu quả của Online-GOPE so với các thuật toán học khác sử dụng cho mô hình LDA.

3.4.1. Các bộ dữ liệu thực nghiệm

Chúng tôi tiến hành thực nghiệm cho các cải tiến trên hai bộ dữ liệu lớn bao gồm các tập văn bản dài: bộ dữ liệu New York Times (NYT) bao gồm 300.000 bài tin tức và bộ PubMed (PUB) bao gồm 330.000 bài báo từ trung tâm PubMed[1]. Chi tiết của hai bộ dữ liệu được mô tả trong Bảng 2.1 đã được trình bày trong Mục 2.4.1 trong Chương 2.

3.4.2. Độ đo đánh giá thực nghiệm

Đánh giá chất lượng của các phương pháp học trong LDA, chúng tôi sử dụng hai độ đo thường được dùng trong mô hình chủ đề, đó là *Log Predictive Probability (LPP)* [93] và *Normalised Pointwise Mutual Information (NPMI)* [102]. Chi tiết về độ đo LPP và NPMI được trình bày trong Phụ lục A và B.

[1]Các bộ dữ liệu được lấy từ http://archive.ics.uci.edu/ml/datasets

3.4.3. Thiết lập các tham số

- **Tham số mô hình:** Chúng tôi thiết lập số chủ đề $K = 100$, tham số Dirichlet $\alpha = \frac{1}{K}$ và siêu tham số $\eta = \frac{1}{K}$. Các tham số này thường được sử dụng trong các mô hình chủ đề.
- **Tham số suy diễn:** Chúng tôi lựa chọn số bước lặp của thuật toán suy diễn $T = 50$ và tham số Bernoulli p trong thuật toán GOPE được lựa chọn trong tập $\{0.10, 0.15, \ldots, 0.85, 0.90\}$ cho mỗi bộ dữ liệu và trên mỗi độ đo.
- **Tham số học:** Chúng tôi lựa chọn kích thước mini-batch $S = |C_t| = 5000$, thiết lập siêu tham số $\kappa = 0.9$ và $\tau = 1$ thích nghi tốt cho các phương pháp suy luận hiện có.

Chúng tôi tiến hành hai nhóm thực nghiệm:

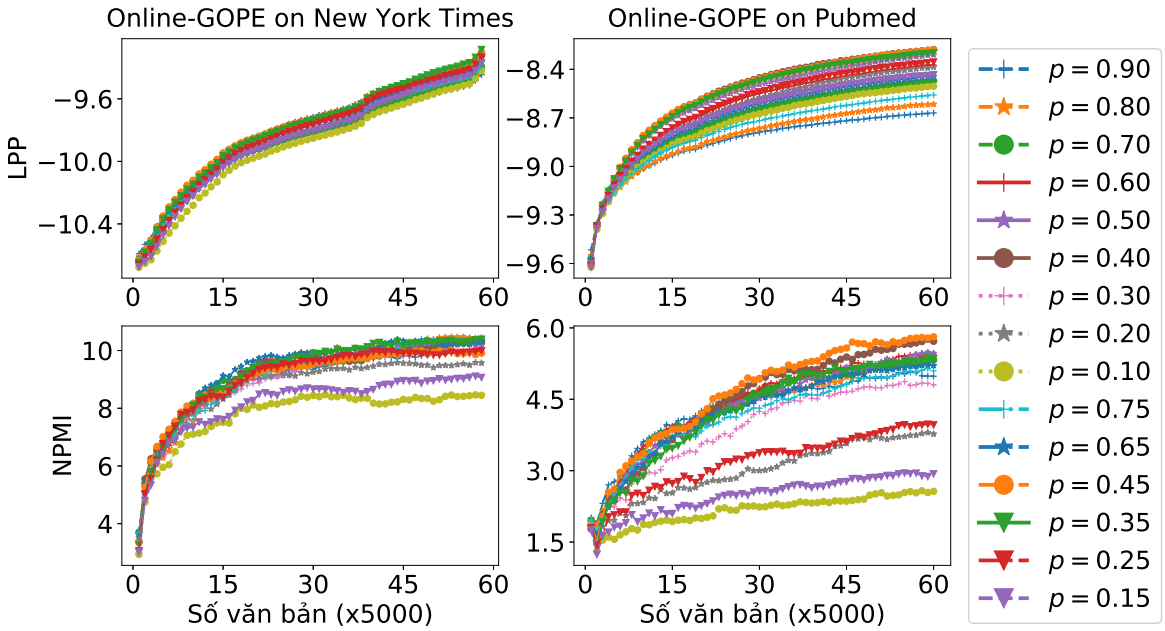
(i) Xem xét vai trò của tham số Bernoulli p trong thuật toán suy diễn GOPE thông qua thuật toán ngẫu nhiên Online-GOPE học mô hình LDA;

(ii) So sánh hiệu quả của thuật toán GOPE so với các thuật toán suy diễn khác như VB, CVB hay CGS thông qua so sánh thuật toán học Online-GOPE với các phương pháp học LDA như Online-VB, Online-CVB, Online-CGS và Online-OPE.

3.4.4. Kết quả thực nghiệm

Đầu tiên, chúng tôi xem xét sự ảnh hưởng của tham số p trong GOPE. Bởi vì $p \in (0, 1)$ và tham số Bernoulli p không nên chọn quá gần với 0 và 1, nên trong các thực nghiệm này chúng tôi chọn tham số Bernoulli p rời rạc trong tập $\{0.10, 0.15, \ldots, 0.85, 0.90\}$ và tiến hành thực nghiệm Online-GOPE trên hai bộ dữ liệu New York Times và PubMed. Kết quả thực hiện thuật toán Online-GOPE khi thay đổi tham số p được mô tả trong Hình 3.1.

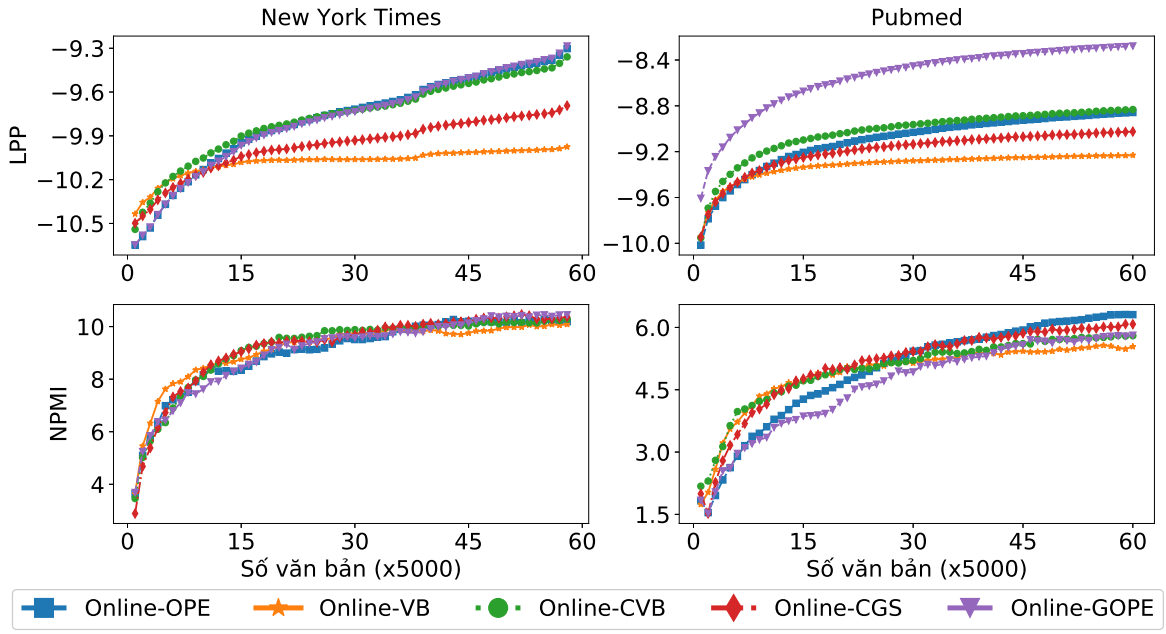
Theo Hình 3.1, chúng ta thấy rằng độ đo LPP và NPMI của thuật toán Online-GOPE thay đổi khi tham số p được lựa chọn khác nhau. Online-GOPE đạt hiệu quả tốt nhất trên bộ New York Times với độ đo LPP khi lựa chọn $p = 0.35$ và với độ đo NPMI khi lựa chọn $p = 0.75$, Online-GOPE đạt hiệu quả tốt nhất trên bộ PubMed với độ đo LPP khi lựa chọn $p = 0.4$, và với độ đo NPMI khi lựa chọn $p = 0.45$. Đồng thời cũng cho thấy, có những giá trị của p làm cho GOPE đạt hiệu quả thấp hơn, do đó cần tránh lựa chọn giá trị p đó khi áp dụng. Kết quả này hỗ trợ cho ý tưởng của chúng tôi về những đóng góp của



Hình 3.1: Kết quả thực hiện Online-GOPE với tham số Bernoulli p được lựa chọn khác nhau trên hai độ đo LPP và NPMI. Giá trị độ đo càng cao càng tốt.

thành phần likelihood và prior trong suy diễn tỷ lệ chủ đề cho một tài liệu. Với các tập dữ liệu khác nhau có giá trị p phù hợp khác nhau, nếu muốn có được hiệu suất tốt nhất về sự khái quát hóa hoặc về chất lượng ngữ nghĩa của các chủ đề, chúng tôi có nhiều giá trị p khác nhau để chọn. Do đó, GOPE rất linh hoạt đối với các bộ dữ liệu trong thế giới thực. Với GOPE, giá trị nào của p là phù hợp tùy thuộc vào thành phần likelihood và prior chiếm tỷ lệ bao nhiêu trong hàm tổng, mà thành phần likelihood phụ thuộc vào độ dài của văn bản. Trong hai bộ dữ liệu thực nghiệm của chúng tôi, độ dài trung bình của một văn bản trong bộ New York Times khoảng hơn 300 từ, trong khi độ dài trung bình của văn bản trong bộ PubMed khoảng 65 từ. Điều đó giải thích tại sao có các giá trị tốt nhất khác nhau của p cho mỗi tập dữ liệu khác nhau. Một điểm thú vị mà tham số Bernoulli p đưa tới chính là nó đóng vai trò như tham số hiệu chỉnh của thuật toán. Các thực nghiệm trên cho thấy ảnh hưởng của giá trị p lựa chọn đến kết quả bài toán. Từ đó, chúng tôi khuyến cáo với mỗi bộ dữ liệu có những đặc trưng khác nhau thì có thể phải lựa chọn p phù hợp khác nhau mới đạt hiệu quả tốt nhất.

Trong các thực nghiệm tiếp theo, chúng tôi so sánh kết quả thực hiện của Online-GOPE với giá trị của p được lựa chọn tốt với các thuật toán khác được thiết kế để học LDA như: Online-VB, Online-CVB0, Online-CGS và Online-OPE. Các kết quả được hiển thị trong Hình 3.2.



Hình 3.2: Kết quả độ đo LPP và NPMI của các thuật toán học Online-OPE, Online-VB, Online-CVB, Online-CGS và Online-GOPE trên hai bộ dữ liệu New York Times và PubMed. Độ đo càng cao càng tốt. Chúng tôi nhận thấy Online-GOPE thường cho kết quả tốt so với các thuật toán học khác.

Các thuật toán học Online-GOPE, Online-OPE, Online-VB, Online-CVB và Online-CGS ở trong Hình 3.2 đều học các chủ đề qua phân phối của các từ $\boldsymbol{\beta}$ hoặc các tham số biến phân $\boldsymbol{\lambda}$. Theo kết quả trong Hình 3.2, chúng tôi thấy Online-GOPE không chỉ thực hiện tốt hơn Online-OPE ban đầu mà còn tốt hơn các phương pháp học đương đại. Chúng tôi có thể lý giải điều này vì Online-GOPE là thuật toán học ngẫu nhiên phù hợp với dữ liệu đầu vào lớn. Sự khác biệt giữa các thuật toán học là do các thủ tục suy diễn bên trong. Online-GOPE hiệu quả chính là do thuật toán suy diễn GOPE tốt hơn các thuật toán suy diễn khác như OPE, VB, CVB hay CGS. Xét về bản chất, GOPE đạt được hiệu quả cao là do sự có mặt của tham số Bernoulli p phù hợp.

3.5. Mở rộng thuật toán giải bài toán tối ưu không lồi

Từ thành công của thuật toán GOPE khi áp dụng cho mô hình chủ đề, chúng tôi nhận thấy hoàn toàn có thể mở rộng thuật toán GOPE cho bài toán tối ưu hóa không lồi (3.2):

$$x^* = \arg\max_x \left[f(x) = g_1(x) + g_2(x) \right]$$

Ý tưởng về cách xây dựng hàm ngẫu nhiên xấp xỉ trong GOPE có thể được sử dụng khi hàm mục tiêu f là tổng của hai thành phần $f = g_1 + g_2$. Trong mỗi bước lặp, chọn thành phần g_1 hay g_2 tuân theo phân phối Bernoulli với xác suất

$p \in (0,1)$. Chúng ta có thể điều chỉnh tham số Bernoulli p phù hợp với các bài toán khác nhau. Tính ngẫu nhiên Bernoulli có thể giúp các thuật toán nhảy ra khỏi cực trị địa phương. Chi tiết của thuật toán GOPE mở rộng cho bài toán không lồi tổng quát được trình bày trong Thuật toán 3.3.

Thuật toán 3.3 GOPE mở rộng cho bài toán không lồi tổng quát

Đầu vào: Tham số Bernoulli $p \in (0,1)$
Đầu ra: $\boldsymbol{x}^*$ là điểm cực đại của hàm $f(\boldsymbol{x}) = g_1(\boldsymbol{x}) + g_2(\boldsymbol{x})$ trên miền Ω

1: Khởi tạo $\boldsymbol{x}_1$ trong miền Ω
2: $G_1 := \frac{g_1}{p}$; $G_2 := \frac{g_2}{1-p}$
3: **for** $t = 1, 2, \ldots, T$ **do**
4: Lấy f_t có phân phối Bernoulli từ $\{G_1, G_2\}$ trong đó

$$\{P(f_t = G_1(\boldsymbol{x})) = p; P(f_t = G_2(\boldsymbol{x})) = 1 - p\}$$

5: $F_t := \frac{1}{t}\sum_{h=1}^{t} f_h$
6: $\boldsymbol{a}_t := \arg\max_{\boldsymbol{x}\in\Omega} \langle F_t'(\boldsymbol{x}_t), \boldsymbol{x}\rangle$
7: $\boldsymbol{x}_{t+1} := \boldsymbol{x}_t + \frac{\boldsymbol{a}_t - \boldsymbol{x}_t}{t}$
8: **end for**

Định lý 3.2 (Sự hội tụ của thuật toán GOPE cho bài toán tối ưu không lồi tổng quát)**.** *Xét hàm mục tiêu $f(\boldsymbol{x})$ trong bài toán* (3.2)*, cho trước tham số Bernoulli $p \in (0,1)$. Xét GOPE, với xác suất 1, ta có:*

(i) Với bất kỳ $\boldsymbol{x} \in \Omega$, dãy hàm $F_t(\boldsymbol{x})$ hội tụ tới $f(\boldsymbol{x})$ khi $t \to +\infty$;

(ii) Dãy nghiệm xấp xỉ $\boldsymbol{x}_t$ hội tụ tới điểm dừng/cực đại địa phương của hàm mục tiêu $f(\boldsymbol{x})$ với tốc độ hội tụ là $\mathcal{O}(1/t)$.

Việc chứng minh Định lý 3.2 tương tự như Định lý 3.1 do quá trình chứng minh không phụ thuộc nhiều vào dạng hàm mục tiêu $f(\boldsymbol{x})$ cụ thể.

3.6. Kết luận chương 3

Trong chương này chúng tôi đã thành công trong việc đề xuất GOPE giải hiệu quả bài toán MAP không lồi trong mô hình chủ đề đảm bảo hội tụ nhanh được đảm bảo bằng cơ sở lý thuyết và thực nghiệm. Hơn nữa, chúng tôi nhận thấy:

- Trong thuật toán GOPE, việc chia hàm mục tiêu f ban đầu thành hai phần g_1 và g_2 tương đối dễ dàng và có thể chia theo nhiều cách. Điều đó thể thuật toán GOPE đảm bảo linh hoạt.
- Cách thức thực hiện GOPE không có quá nhiều ràng buộc trên hàm mục tiêu f nên có thể áp dụng tốt với hàm mục tiêu lồi hoặc không lồi;

- Thuật toán GOPE có thể áp dụng tốt cho bài toán quy hoạch DC có hàm mục tiêu f là hiệu của hai hàm lồi $f = g - h$ vì có thể đặt $g_1 := g$ và $g_2 := -h$, nên hàm f được viết lại dưới dạng $f = g_1 + g_2$ có dạng trong thuật toán GOPE;
- Thuật toán GOPE có thể áp dụng để giải bài toán hiệu chỉnh

$$\boldsymbol{\theta}^* = \arg\min \mathcal{L}(\boldsymbol{\theta}) + \lambda \mathcal{R}(\boldsymbol{\theta})$$

  trong đó $\mathcal{R}(\boldsymbol{\theta})$ là phần hiệu chỉnh, λ là hệ số hiệu chỉnh, thông thường $\lambda \in (0, 1)$ khi đặt $g_1 := \mathcal{L}(\boldsymbol{\theta})$ và $g_2 := \lambda \mathcal{R}(\boldsymbol{\theta})$

Các kết quả trình bày trong chương này được chúng tôi trình bày trong bài báo "A flexible stochastic method for solving the MAP problem in topic models" đăng trên tạp chí quốc tế Computación y Sistemas trong danh mục SCOPUS và ESCI năm 2018.

Chương 4

NGẪU NHIÊN BERNOULLI CHO BÀI TOÁN MAP KHÔNG LỒI VÀ ỨNG DỤNG

Trong chương này, chúng tôi tiếp tục nghiên cứu bài toán ước lượng MAP không lồi trong các mô hình đồ thị xác suất. Chúng tôi sử dụng ngẫu nhiên hóa Bernoulli với xác suất $p \in (0, 1)$ kết hợp với hai biên ngẫu nhiên để thiết kế thuật toán tối ưu ngẫu nhiên BOPE giải hiệu quả bài toán MAP không lồi, từ đó áp dụng thành công thuật toán BOPE vào bài toán phân tích văn bản và bài toán gợi ý.

4.1. Giới thiệu

MAP là phương pháp ước lượng thông dụng trong học máy thống kê [106]. Xét bài toán MAP có dạng sau:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} [\log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})] \tag{4.1}$$

trong đó $P(D|\boldsymbol{x})$ ký hiệu là likelihood của quan sát D, $P(\boldsymbol{x})$ chính là prior của biến ẩn $\boldsymbol{x}$ và $P(D)$ là xác suất biên của D. Hàm mục tiêu của bài toán MAP chính là:

$$f(\boldsymbol{x}) = \log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})$$

Từ (4.1) chúng ta thấy ước lượng MAP có vai trò của một kỹ thuật hiệu chỉnh. Hàm mục tiêu của bài toán gồm hai thành phần: phần $\log P(D|\boldsymbol{x})$ được xem như hàm mất mát chính, còn $\log P(\boldsymbol{x})$ được coi như thành phần hiệu chỉnh. Bên cạnh những lợi thế của ước lượng MAP như giúp tránh hiện tượng quá khớp, chúng ta có thể phải đối mặt với khó khăn khi bài toán MAP là không lồi và không khả thi trong trường hợp xấu [37].

Chúng tôi nhận thấy có một số phương pháp xấp xỉ để giải bài toán suy diễn như Variational Bayes (VB) [39], collapsed Variational Bayes (CVB) [40, 41], CVB0 [42], Collapsed Gibbs Sampling (CGS) [43], Concave-Convex procedure (CCCP) [44], Stochastic Majorization-Minimization (SMM) [45], Particle Mirror Descent (PMD) [49], HAMCMC [50], Block-coordinate Frank-Wolfe [48]. Đây có thể được xem như các phương pháp suy diễn tiên tiến. Tuy nhiên, chúng tôi cũng nhìn thấy một số phương pháp mới chỉ được thiết kế để sử dụng cho một số mô hình cụ thể [39, 46], hoặc chúng chưa đảm bảo các tiêu chuẩn về sự hội tụ, tốc độ hội tụ, khả năng linh hoạt và khả năng hiệu chỉnh.

Đóng góp của luận án là đề xuất một thuật toán tối ưu ngẫu nhiên BOPE thông qua sử dụng ngẫu nhiên Bernoulli và hai biên ngẫu nhiên. Theo hiểu biết của chúng tôi, bài toán MAP trong nhiều mô hình xác suất trên thực tế có bản chất không lồi, do đó nó thuộc lớp bài toán NP-khó [37]. Thuật toán mới của chúng tôi có bản chất ngẫu nhiên và về mặt lý thuyết hội tụ đến một điểm cực đại/điểm dừng địa phương của MAP. BOPE có thể dễ dàng được sử dụng trong các bối cảnh như tối ưu không lồi. Với ưu thế này, BOPE khắc phục được nhiều nhược điểm của các phương pháp suy diễn hiện có và vượt qua các tiêu chí như hội tụ, ngẫu nhiên, tốc độ hội tụ, linh hoạt hoặc chính quy. Kết quả là, một khung chung để giải quyết các vấn đề MAP không lồi được đề xuất. Hiệu quả của BOPE đã được nghiên cứu trên hai khía cạnh lý thuyết và thực nghiệm. Chúng tôi chứng minh rằng BOPE hội tụ với $\mathcal{O}(1/T)$, đây là tốc độ hội tụ tốt nhất cho bài toán MAP hiện tại. Ngoài ra, chúng tôi trình bày cách sử dụng BOPE trong bối cảnh rộng hơn, bao gồm tối ưu hóa không lồi. Chúng tôi cũng phát hiện ra rằng BOPE có vai trò của một phương pháp hiệu chỉnh. Chúng ta sử dụng BOPE là thuật toán suy diễn để thiết kế phương pháp ngẫu nhiên Online-BOPE học các mô hình chủ đề ở quy mô lớn. Hiệu quả của thuật toán BOPE về mặt thực nghiệm được chúng tôi chứng minh thông qua triển khai ứng dụng BOPE vào hai bài toán phân tích văn bản với các bộ dữ liệu lớn và bài toán hệ gợi ý. Hơn nữa, chúng tôi tin tưởng rằng với các ưu việt của BOPE, chúng tôi có thể áp dụng rộng rãi BOPE vào giải quyết cho các bài toán không lồi phức tạp khác xuất hiện trong học máy.

4.2. Thuật toán BOPE giải bài toán MAP không lồi

4.2.1. Ý tưởng xây dựng thuật toán BOPE

Trong phần này, chúng tôi giới thiệu thuật toán BOPE để giải bài toán MAP (4.1) trong đó mục tiêu $f(\boldsymbol{x})$ là hàm trơn và không lồi trên miền đóng Ω:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}\in\Omega}[\log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})]$$

Ý tưởng của BOPE khá đơn giản. BOPE đề xuất để giải bài toán (4.1) theo tư tưởng của phương pháp lặp. Khi số lần lặp tiến đến vô cùng, BOPE sẽ tiệm cận đến điểm cực đại/điểm dừng cục bộ của bài toán (4.1). Chi tiết về BOPE được trình bày trong Thuật toán 4.1.

Thuật toán BOPE thu được bằng cách phân tích hàm mục tiêu $f(\boldsymbol{x})$ ban đầu thành hai phần g_1 và g_2, sau đó sử dụng phân phối Bernoulli để xây dựng hai dãy biên ngẫu nhiên của hàm mục tiêu. Ký hiệu $g_1(\boldsymbol{x}) = \log P(D|\boldsymbol{x})$ và $g_2(\boldsymbol{x}) = \log P(\boldsymbol{x})$. Giả sử rằng $g_1(\boldsymbol{x})$ và $g_2(\boldsymbol{x})$ có đạo hàm liên tục trên miền Ω. Chúng tôi sử dụng phân phối Bernoulli với tham số $p \in (0, 1)$ thay thế cho phân

Thuật toán 4.1 BOPE giải bài toán MAP không lồi

Đầu vào: Tham số Bernoulli $p \in (0,1)$
Đầu ra: $\boldsymbol{x}^*$ là điểm cực đại của hàm số $f(\boldsymbol{x}) = \log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})$ trên miền Ω

Khởi tạo $\boldsymbol{x}_1$ trong Ω
$G_1(\boldsymbol{x}) := \frac{\log P(D|\boldsymbol{x})}{p}$; $G_2(\boldsymbol{x}) := \frac{\log P(\boldsymbol{x})}{1-p}$
$f_1^l := G_1(\boldsymbol{x})$ và $f_1^u := G_2(\boldsymbol{x})$
for $t = 2, 3, \ldots, \infty$ **do**
    Lấy f_t^l có phân phối Bernoulli từ $\{G_1(\boldsymbol{x}), G_2(\boldsymbol{x})\}$ trong đó

$$P(f_t^l = G_1(\boldsymbol{x})) = p; P(f_t^l = G_2(\boldsymbol{x})) = 1 - p$$

    $L_t := \frac{1}{t}\sum_{h=1}^{t} f_h^l$
    $\boldsymbol{a}_t^l := \arg\max_{\boldsymbol{x}\in\Omega} < L_t'(\boldsymbol{x}_t), \boldsymbol{x} >$
    $\boldsymbol{x}_{t+1}^l := \boldsymbol{x}_t + \frac{\boldsymbol{a}_t^l - \boldsymbol{x}_t}{t}$
    Lấy f_t^u có phân phối Bernoulli từ $\{G_1(\boldsymbol{x}),\ G_2(\boldsymbol{x})\}$ trong đó

$$P(f_t^u = G_1(\boldsymbol{x})) = p; P(f_t^u = G_2(\boldsymbol{x})) = 1 - p$$

    $U_t := \frac{1}{t}\sum_{h=1}^{t} f_h^u$
    $\boldsymbol{a}_t^u := \arg\max_{\boldsymbol{x}\in\Omega} < U_t'(\boldsymbol{x}_t), \boldsymbol{x} >$
    $\boldsymbol{x}_{t+1}^u := \boldsymbol{x}_t + \frac{\boldsymbol{a}_t^u - \boldsymbol{x}_t}{t}$
    $\boldsymbol{x}_{t+1} := \arg\max_{\boldsymbol{x}\in\{\boldsymbol{x}_{t+1}^u,\ \boldsymbol{x}_{t+1}^l\}} f(\boldsymbol{x})$
end for

phối đều đơn giản trong OPE và xây dựng hai dãy xấp xỉ cho hàm mục tiêu đúng $f(\boldsymbol{x})$, trong đó một dãy xuất phát từ $g_1(\boldsymbol{x})$, gọi là dãy cận dưới $\{L_t\}$, một dãy khác xuất phát từ $g_2(\boldsymbol{x})$ gọi là dãy cận trên $\{U_t\}$. Với tham số Bernoulli $p \in (0,1)$ cho trước, tiến hành hiệu chỉnh likelihood g_1 và prior g_2 như sau:

$$G_1(\boldsymbol{x}) := \frac{g_1(\boldsymbol{x})}{p};\ G_2(\boldsymbol{x}) := \frac{g_2(\boldsymbol{x})}{1-p}$$

Khởi tạo $f_1^l := G_1(\boldsymbol{x})$. Với mỗi bước lặp t $(t = 1, 2, .., T)$, lấy f_t^l tuân theo phân phối Bernoulli với xác suất $p \in (0,1)$ từ tập $\{G_1(\boldsymbol{x}),\ G_2(\boldsymbol{x})\}$ trong đó

$$P(f_t^l = G_1(\boldsymbol{x})) = p,\ P(f_t^l = G_2(\boldsymbol{x})) = 1 - p,\ t = 2, 3, \ldots$$

Thiết lập dãy

$$L_t := \frac{1}{t}\sum_{h=1}^{t} f_h^l$$

và giải bài toán quy hoạch tuyến tính trên Ω:

$$\boldsymbol{a}_t^l := \arg\max_{\boldsymbol{x}\in\Omega} < L_t'(\boldsymbol{x}_t), \boldsymbol{x} >$$

sau đó xây dựng sơ đồ lặp:

$$\boldsymbol{x}_{t+1}^l := \boldsymbol{x}_t + \frac{\boldsymbol{a}_t^l - \boldsymbol{x}_t}{t}$$

Tiếp theo, xây dựng dãy $\{U_t\}$ tương tự như dãy $\{L_t\}$. Thiết lập khởi tạo $f_1^u := G_2(\boldsymbol{x})$. Với mỗi bước lặp t $(t = 1, 2, .., T)$, lấy f_t^u có phân phối Bernoulli với xác

suất $p \in (0,1)$ từ tập $\{G_1(\boldsymbol{x}),\ G_2(\boldsymbol{x})\}$ trong đó

$$P(f_t^u = G_1(\boldsymbol{x})) = p\ ,\ P(f_t^u = G_2(\boldsymbol{x})) = 1-p,\ t = 2,3,\ldots$$

Khi đó thu được

$$U_t := \frac{1}{t}\sum_{h=1}^{t} f_h^u$$

và giải bài toán quy hoạch tuyến tính trên Ω:

$$\boldsymbol{a}_t^u := \arg\max_{\boldsymbol{x}\in\Omega} < U_t'(\boldsymbol{x}_t), \boldsymbol{x} >$$

xây dựng lược đồ lặp

$$\boldsymbol{x}_{t+1}^u := \boldsymbol{x}_t + \frac{\boldsymbol{a}_t^u - \boldsymbol{x}_t}{t}$$

Dễ dàng nhận thấy L_t và U_t chính là trung bình của các mẫu ngẫu nhiên và chúng đều đảm bảo hội tụ đến hàm mục tiêu đúng $f(\boldsymbol{x})$ khi $t \to \infty$. Điều này được chỉ rõ trong quá trình chứng minh của Định lý 4.1.

Mặt khác, khám phá thấy rằng tham số Bernoulli p đóng vai trò điều chỉnh tỷ lệ likelihood và prior đóng góp vào dãy L_t và U_t. Tại mỗi bước lặp, sử dụng hai dãy ngẫu nhiên $\{L_t\}$ và $\{U_t\}$ giúp có nhiều thông tin hơn về hàm mục tiêu $f(\boldsymbol{x})$, vì vậy chúng ta có cơ hội tìm đến nghiệm tối ưu của $f(\boldsymbol{x})$ nhanh hơn. Với dãy $\{L_t\}$ thông qua lược đồ lặp thu được dãy nghiệm xấp xỉ $\{\boldsymbol{x}_t^l\}$, còn với dãy $\{U_t\}$ thu được dãy nghiệm xấp xỉ $\{\boldsymbol{x}_t^u\}$. Khi đó, trong mỗi bước lặp, sử dụng nguyên lý tham lam, chúng ta so sánh hai giá trị $f(\boldsymbol{x}_t^u)$ và $f(\boldsymbol{x}_t^l)$ của hàm mục tiêu và lựa chọn điểm làm cho hàm mục tiêu $f(\boldsymbol{x})$ đạt giá trị cao nhất là nghiệm xấp xỉ tại bước lặp thứ t:

$$\boldsymbol{x}_{t+1} := \arg\max_{\boldsymbol{x}\in\{\boldsymbol{x}_{t+1}^u, \boldsymbol{x}_{t+1}^l\}} f(\boldsymbol{x})$$

Thuật toán BOPE sử dụng phân phối Bernoulli tổng quát hơn phân phối đều và đồng thời tạo ra ba dãy số $\{\boldsymbol{x}_t^u\}$, $\{\boldsymbol{x}_t^l\}$ và $\{\boldsymbol{x}_t\}$ phụ thuộc lẫn nhau để tìm đến nghiệm tối ưu $\{\boldsymbol{x}_t\}$ tại bước lặp thứ t.

4.2.2. Sự hội tụ của thuật toán BOPE

Định lý 4.1 (Sự hội tụ của BOPE)**.** *Giả sử rằng $g_1(\boldsymbol{x})$ và $g_2(\boldsymbol{x})$ có đạo hàm liên tục trên miền đóng Ω. Cho trước tham số Bernoulli $p \in (0,1)$, với xác suất 1, dãy nghiệm $\{\boldsymbol{x}_t\}$ thu được bởi Thuật toán 4.1 đảm bảo hội tụ đến điểm cực đại địa phương hoặc điểm dừng $\boldsymbol{x}^*$ của hàm mục tiêu $f(\boldsymbol{x})$ với tốc độ hội tụ $\mathcal{O}(1/T)$ trong đó T là số bước lặp thực hiện.*

Chứng minh. Giả sử rằng hàm mục tiêu $f(\boldsymbol{x})$ là không lồi trên miền ràng buộc Ω. Theo hiểu biết của chúng tôi, tiêu chuẩn được sử dụng cho phân tích hội tụ

của các thuật toán tối ưu là rất quan trọng đối với tối ưu không lồi. Đối với các bài toán tối ưu không ràng buộc, $\|\nabla f(\boldsymbol{x})\|$ thường được sử dụng để đánh giá sự hội tụ, bởi vì $\|\nabla f(\boldsymbol{x})\| \to 0$ dẫn đến hội tụ tới một điểm dừng. Tuy nhiên, tiêu chí này không sử dụng được cho các bài toán tối ưu không lồi có ràng buộc. Thay vào đó, chúng tôi sử dụng tiêu chuẩn "Frank-Wolfe gap" trong [107]. Tiến hành hiệu chỉnh thành phần likelihood và prior theo tham số Bernoulli p tương ứng như sau:

$$G_1(\boldsymbol{x}) := g_1(\boldsymbol{x})/p \ ; \ G_2(\boldsymbol{x}) := g_2(\boldsymbol{x})/1-p$$

Khi đó

$$f(\boldsymbol{x}) = g_1(\boldsymbol{x}) + g_2(\boldsymbol{x}) = p.G_1(\boldsymbol{x}) + (1-p)G_2(\boldsymbol{x})$$

Đầu tiên, xem xét dãy $\{U_t\}$. Đặt $f_1^u := g_2(\boldsymbol{x})$. Với $t = 2, 3, \ldots, \infty$, f_t^u có phân phối Bernoulli từ $\{G_1(\boldsymbol{x}), G_2(\boldsymbol{x})\}$ trong đó

$$\{P(f_t^u = G_1(\boldsymbol{x})) = p; P(f_t^u = G_2(\boldsymbol{x})) = 1-p\}$$

và nhận được $U_t := \frac{1}{t}\sum_{h=1}^{t} f_h^u$. Gọi a_t và $b_t = t - a_t$ là số lần lấy được thành phần $G_1(\boldsymbol{x})$ và $G_2(\boldsymbol{x})$ tương ứng sau t bước lặp. Do đó, ta có:

$$U_t = \frac{1}{t}(a_t G_{_1} + (t-a_t)G_2) \tag{4.2}$$

Nhận thấy a_t là một biến ngẫu nhiên có phân phối nhị thức với tham số t và p, tức là $a_t \sim B(t,p)$. Khi đó kỳ vọng $E[a_t] = tp$ và phương sai $D[a_t] = tp(1-p)$. Theo định lý Moivre-Laplace [105, 108], khi $t \to \infty$ thì a_t xấp xỉ phân phối chuẩn với kỳ vọng tp và phương sai $tp(1-p)$ hay $a_t \to N(tp, tp(1-p))$.
Đặt $S_t = a_t - tp$. Ta có $S_t \to N(0, tp(1-p))$ khi $t \to \infty$. Vì vậy, $\frac{S_t}{t} \to 0$ khi $t \to \infty$ với xác suất 1.

$$U_t - f = \frac{S_t}{t}(G_1 - G_2) \tag{4.3}$$

$$U_t' - f' = \frac{S_t}{t}(G_1' - G_2') \tag{4.4}$$

Do vậy, U_t là một ước lượng không chệch của $f(\boldsymbol{x})$. Ta có $\frac{S_t}{t} \to 0$ khi $t \to +\infty$. Kết hợp với (4.3), thu được dãy $U_t \to f$ với xác suất 1. Từ (4.4), ta có được dãy đạo hàm $U_t' \to f'$ khi $t \to +\infty$. Sự hội tụ này có được với bất kì $\boldsymbol{x} \in \overline{\Omega}$.

Xem xét

$$\begin{aligned}\langle U_t'(\boldsymbol{x}_t), \frac{\boldsymbol{a}_t^u - \boldsymbol{x}_t}{t}\rangle &= \langle U_t'(\boldsymbol{x}_t) - f'(\boldsymbol{x}_t), \frac{\boldsymbol{a}_t^u - \boldsymbol{x}_t}{t}\rangle + \langle f'(\boldsymbol{x}_t), \frac{\boldsymbol{a}_t^u - \boldsymbol{x}_t}{t}\rangle \\ &= \frac{S_t}{t^2}\langle G_1'(\boldsymbol{x}_t) - G_2'(\boldsymbol{x}_t), \boldsymbol{a}_t^u - \boldsymbol{x}_t\rangle + \langle f'(\boldsymbol{x}_t), \frac{\boldsymbol{a}_t^u - \boldsymbol{x}_t}{t}\rangle\end{aligned}$$

Nhận thấy g_1 và g_2 là Lipschitz liên tục trên miền Ω. Vì vậy, tồn tại một hằng số L sao cho

$$\langle f'(z), y - z\rangle \leq f(y) - f(z) + L\|y - z\|^2, \ \forall \ y, z \in \Omega$$

$$\langle f'(\boldsymbol{x}_t), \frac{\boldsymbol{a}_t^u - \boldsymbol{x}_t}{t} \rangle = \langle f'(\boldsymbol{x}_t), \boldsymbol{x}_{t+1}^u - \boldsymbol{x}_t \rangle$$
$$\leq f(\boldsymbol{x}_{t+1}^u) - f(\boldsymbol{x}_t) + L\|\boldsymbol{x}_{t+1}^u - \boldsymbol{x}_t\|^2 = f(\boldsymbol{x}_{t+1}^u) - f(\boldsymbol{x}_t) + L\|\frac{\boldsymbol{a}_t^u - \boldsymbol{x}_t}{t}\|^2$$

Ta có $\boldsymbol{x}_{t+1} := \arg\max_{\boldsymbol{x} \in \{\boldsymbol{x}_{t+1}^u, \boldsymbol{x}_{t+1}^l\}} f(\boldsymbol{x})$, vì vậy

$$f(\boldsymbol{x}_{t+1}^u) \leq f(\boldsymbol{x}_{t+1})$$

Vì $\boldsymbol{a}_t^u$ và $\boldsymbol{x}_t$ thuộc vào miền Ω, nên $|\langle G_1'(\boldsymbol{x}_t) - G_2'(\boldsymbol{x}_t), \boldsymbol{a}_t^u - \boldsymbol{x}_t \rangle|$ và $\|\boldsymbol{a}_t^u - \boldsymbol{x}_t\|^2$ bị chặn với bất kỳ t nào. Do vậy, tồn tại một hằng số $c_1 > 0$ sao cho

$$\langle U_t'(\boldsymbol{x}_t), \frac{\boldsymbol{a}_t^u - \boldsymbol{x}_t}{t} \rangle \leq c_1 \frac{|S_t|}{t^2} + f(\boldsymbol{x}_{t+1}) - f(\boldsymbol{x}_t) + \frac{c_1 L}{t^2} \tag{4.5}$$

Lấy tổng hai vế của (4.5) với mọi t, thu được

$$\sum_{t=1}^{+\infty} \frac{1}{t} \langle U_t'(\boldsymbol{x}_t), \boldsymbol{a}_t^u - \boldsymbol{x}_t \rangle \leq \sum_{t=1}^{+\infty} c_1 \frac{|S_t|}{t^2} + f(\boldsymbol{x}_{+\infty}) - f(\boldsymbol{x}_1) + \sum_{t=1}^{+\infty} \frac{c_1 L}{t^2} \tag{4.6}$$

Bởi vì $f(\boldsymbol{x})$ bị chặn nên $f(\boldsymbol{x}_{+\infty})$ cũng bị chặn. Mặt khác $S_t = \mathcal{O}(\sqrt{t \log t})$ theo [103], vì vậy $\sum_{t=1}^{+\infty} c_1 \frac{|S_t|}{t^2}$ hội tụ với xác suất 1 và $\sum_{t=1}^{+\infty} \frac{L}{t^2}$ cũng bị chặn. Do vậy, vế phải của (4.6) là hữu hạn. Ngoài ra, $\langle U_t'(\boldsymbol{x}_t), \boldsymbol{a}_t^u \rangle > \langle U_t'(\boldsymbol{x}_t), \boldsymbol{x}_t \rangle$ với bất kỳ $t > 0$ bởi vì $a_t^u = \arg\max_{x \in \Omega} \langle U_t'(\boldsymbol{x}_t), \boldsymbol{x} \rangle$. Vì vậy, chúng ta thu được:

$$0 \leq \sum_{t=1}^{+\infty} \frac{1}{t} \langle U_t'(\boldsymbol{x}_t), \boldsymbol{a}_t^u - \boldsymbol{x}_t \rangle < \infty \tag{4.7}$$

Nói cách khác, chuỗi tổng $\sum_{t=1}^{+\infty} \frac{1}{t} \langle U_t'(\boldsymbol{x}_t), \boldsymbol{a}_t^u - \boldsymbol{x}_t \rangle$ hội tụ tới một hằng hữu hạn. Nhớ rằng $\langle U_t'(\boldsymbol{x}_t), \boldsymbol{a}_t^u - \boldsymbol{x}_t \rangle \geq 0$ với bất kỳ t. Nếu tồn tại hằng số $c_2 > 0$ thỏa mãn $\langle U_t'(\boldsymbol{x}_t), \boldsymbol{a}_t^u - \boldsymbol{x}_t \rangle \geq c_2$ với bất kỳ giá trị nào của t, thì chuỗi $\sum_{t=1}^{+\infty} \frac{1}{t} \langle U_t'(\boldsymbol{x}_t), \boldsymbol{a}_t^u - \boldsymbol{x}_t \rangle$ không thể hội tụ tới hằng hữu hạn. Điều này mâu thuẫn với (4.7). Vì vậy:

$$\langle U_t'(\boldsymbol{x}_t), \boldsymbol{a}_t^u - \boldsymbol{x}_t \rangle \to 0 \text{ as } t \to +\infty \tag{4.8}$$

Bởi vì $U_t' \to f'$ khi $t \to \infty$ và f' là liên tục, kết hợp với (4.8) ta có

$$\langle f'(\boldsymbol{x}_t), \boldsymbol{a}_t^u - \boldsymbol{x}_t \rangle \to 0 \text{ khi } t \to +\infty \tag{4.9}$$

Áp dụng tiêu chuẩn "Frank-Wolfe gap", từ (4.9) thu được $\boldsymbol{x}_t \to \boldsymbol{x}^*$ khi $t \to +\infty$. Nói cách khác, $\boldsymbol{x}_t$ hội tụ theo xác suất tới điểm dừng $\boldsymbol{x}^*$ của hàm mục tiêu $f(\boldsymbol{x})$. □

Chúng tôi đã chỉ ra rằng BOPE và OPE ít nhất hội tụ với tốc độ $\mathcal{O}(1/T)$ trong khi PMC hội tụ với $\mathcal{O}(T^{-1/2})$ và HAMCMC hội tụ với tốc độ $\mathcal{O}(T^{-1/3})$ trong đó T là số lần lặp. Hơn nữa, BOPE tổng quát và linh hoạt hơn OPE thông qua sử dụng ngẫu nhiên Bernoulli, tức là khi thay đổi giá trị tham số Bernoulli p, chúng

tôi nhận được các biến thể khác nhau của BOPE. Khi đó tham số Bernoulli p giúp điều chỉnh sự đóng góp của likelihood hay prior trong BOPE. Như vậy, có thể căn cứ vào mô hình và đặc điểm dữ liệu để lựa chọn giá trị tham số p phù hợp để thuật toán đạt hiệu quả tốt nhất.

4.2.3. Vai trò hiệu chỉnh của thuật toán BOPE

Trong học máy, khi xây dựng mô hình hoặc tối ưu tham số mô hình, chúng ta thường hay gặp hiện tượng quá khớp. Đó là hiện tượng mô hình tìm được quá khớp với dữ liệu huấn luyện. Việc quá khớp này có thể dẫn đến việc dự đoán nhầm và chất lượng mô hình không còn tốt trên dữ liệu tương lai. Về cơ bản, hiện tượng quá khớp xảy ra khi mô hình quá phức tạp để mô phỏng dữ liệu huấn luyện. Điều này đặc biệt xảy ra khi lượng dữ liệu huấn luyện quá nhỏ trong khi độ phức tạp của mô hình quá cao. Một trong những phương pháp đơn giản để làm giảm hoặc tránh hiện tượng quá khớp hiệu quả chính là kỹ thuật hiệu chỉnh, tức là thay đổi mô hình một chút để tránh hiện tượng quá khớp xảy ra và cải thiện tính tổng quát của nó. Giả sử cần tối ưu hàm đánh giá $f(\boldsymbol{x})$ theo bài toán sau:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} f(\boldsymbol{x})$$

Khi bài toán đặt ở điều kiện xấu hoặc có thể xảy ra hiện tượng quá khớp, chúng ta có thể sử dụng kỹ thuật hiệu chỉnh bằng cách bổ sung thêm một thành phần hiệu chỉnh $R(\boldsymbol{x})$ vào hàm mục tiêu $f(\boldsymbol{x})$ ban đầu:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} [f(\boldsymbol{x}) + \lambda R(\boldsymbol{x})]$$

trong đó λ là tham số hiệu chỉnh.

Chúng tôi thấy rằng ước lượng MAP đóng vai trò là một phương pháp hiệu chỉnh của ước lượng MLE, bởi vì trong hàm mục tiêu của MAP đã cộng thêm thành phần prior $\log P(\boldsymbol{x})$ của biến ẩn $\boldsymbol{x}$ vào thành phần likelihood $\log P(D|\boldsymbol{x})$ của quan sát D:

$$x^* = \arg\max_{\boldsymbol{x}} [\log P(D|\boldsymbol{x}) + \log P(\boldsymbol{x})]$$

Do đó MAP hiệu quả hơn MLE và khắc phục được hiện tượng quá khớp và dữ liệu ít của phương pháp MLE. Thiết kế thuật toán ngẫu nhiên BOPE giải bài toán MAP không lồi được trình bày trong Thuật toán 4.1, chúng tôi khám phá ra vai trò quan trọng của tham số Bernoulli $p \in (0, 1)$ góp phần đưa BOPE trở thành một thuật toán có khả năng hiệu chỉnh. Đây là một lợi thế nổi bật và khác biệt của BOPE mà không nhìn thấy trong các thuật toán trước đó. Khả năng hiệu chỉnh tốt của BOPE được chúng tôi làm rõ trên hai phương diện lý thuyết và thực nghiệm. Về mặt lý thuyết, tính hiệu chỉnh của BOPE được chúng tôi làm rõ thông qua Định lý 4.2.

Đặt $g_1(\boldsymbol{x}) = \log P(D|\boldsymbol{x}); g_2(\boldsymbol{x}) = \log P(\boldsymbol{x}$. Khi đó bài toán MAP không lồi (4.1) có dạng:

$$x^* = \arg\max_{\boldsymbol{x}}[f(\boldsymbol{x}) = g_1(\boldsymbol{x}) + g_2(\boldsymbol{x})]$$

Định lý 4.2 (Tính hiệu chỉnh của BOPE)**.** *Giả sử cho trước tham số Bernoulli $p \in (0,1)$, xét thuật toán BOPE giải bài toán MAP không lồi* (4.1)*. Khi đó, thuật toán BOPE đưa về tối ưu hàm mục tiêu mới có dạng $f(\boldsymbol{x}) + R(g_1, g_2, p)$ với $R(g_1, g_2, p) = h(t,p)(\frac{g_1(\boldsymbol{x})}{p} - \frac{g_2(\boldsymbol{x})}{1-p})$, trong đó $h(t,p) \to 0$ khi số vòng lặp $t \to \infty$. Như vậy, BOPE là một kỹ thuật hiệu chỉnh với $R(g_1, g_2, p)$ là thành phần hiệu chỉnh và tham số Bernoulli p là tham số hiệu chỉnh.*

Chứng minh. Xem xét BOPE được trình bày trong Thuật toán 4.1. Xét dãy xấp xỉ ngẫu nhiên $\{U_t\}$) được xây dựng như sau:

1. Cho giá trị tham số Bernoulli $p \in (0,1)$. Tiến hành hiệu chỉnh như sau: $G_1(\boldsymbol{x}) = \frac{1}{p} g_1(\boldsymbol{x})$ và $G_2(\boldsymbol{x}) = \frac{1}{1-p} g_2(\boldsymbol{x})$.
2. Lấy f_t^u là biến ngẫu nhiên tuân theo phân phối Bernoulli với xác suất $p \in (0,1)$ cho trước, lựa chọn từ tập $\{G_1(\boldsymbol{x}),\ G_2(\boldsymbol{x})\}$ trong đó

$$P(f_t^u = G_1(\boldsymbol{x})) = p\ ,\ P(f_t^u = G_2(\boldsymbol{x})) = 1 - p,\ t = 2, 3, \ldots$$

3. Khi đó thu được xấp xỉ cho hàm mục tiêu $f(\boldsymbol{x})$ là dãy ngẫu nhiên

$$U_t := \frac{1}{t} \sum_{h=1}^{t} f_h^u$$

Như vậy, bản chất U_t là trung bình của t biến ngẫu nhiên $f_1^u, f_2^u, .., f_t^u$.

Đặt a_t và b_t tương ứng là số lần xuất hiện thành phần $G_1(\boldsymbol{x})$ và $G_2(\boldsymbol{x})$ trong dãy hàm U_t sau t bước lặp. Khi đó $a_t + b_t = t$ và $U_t = \frac{1}{t}(a_t G_1 + b_t G_2)$. Ta có a_t có phân phối nhị thức với tham số t và p. Hơn nữa, theo luật số lớn, ta có a_t xấp xỉ phân phối chuẩn với kỳ vọng $E[a_t] = tp$ và phương sai $D[a_t] = tp(1-p)$ khi $t \to +\infty$. Thay vì tiến hành tối ưu tham số trực tiếp trên hàm mục tiêu đúng $f(\boldsymbol{x})$, thuật toán BOPE tiến hành cực đại hóa hàm xấp xỉ $U_t(\boldsymbol{x})$ (tương tự cực đại hóa hàm $L_t(\boldsymbol{x})$).

Đặt $S_t = a_t - tp$ và $f(\boldsymbol{x}) = g_1(\boldsymbol{x}) + g_2(\boldsymbol{x}) = pG_1(\boldsymbol{x}) + (1-p)G_2(\boldsymbol{x})$, dãy hàm $U_t(\boldsymbol{x})$ được viết lại như sau

$$U_t(\boldsymbol{x}) = f(\boldsymbol{x}) + \frac{S_t}{t}(G_1(\boldsymbol{x}) - G_2(\boldsymbol{x})) \tag{4.10}$$

Theo phương trình (4.10), hàm U_t là tổng của hàm mục tiêu đúng $f(\boldsymbol{x})$ và thành phần $\frac{S_t}{t}(G_1(\boldsymbol{x}) - G_2(\boldsymbol{x}))$. Như vậy, hàm U_t là hàm mục tiêu hiệu chỉnh, còn thành phần $\frac{S_t}{t}(G_1(\boldsymbol{x}) - G_2(\boldsymbol{x}))$ là phần hiệu chỉnh bổ sung thêm vào. Theo luật LIL

[103] và chứng minh của Định lý 4.1, chúng tôi thu được $\frac{S_t}{t} \to 0$ khi $t \to \infty$ với xác suất 1, nên thành phần $\frac{S_t}{t}(G_1(\boldsymbol{x}) - G_2(\boldsymbol{x}))$ đảm bảo tiến về 0 khi $t \to \infty$.

Theo cách thức xây dựng thuật toán BOPE, chúng tôi có:

$$\frac{S_t}{t} = \frac{a_t - tp}{t} = \frac{a_t}{t} - p$$

trong đó $E[\frac{a_t}{t}] = p$ và $D[\frac{a_t}{t}] = \frac{p(1-p)}{t}$. Khi đó $E[\frac{S_t}{t}] = 0$ và $D[\frac{S_t}{t}] = \frac{p(1-p)}{t} \to 0$ khi $t \to \infty$. Ngoài ra, khảo sát hàm $g(p) = p(1-p)$ với $p \in (0,1)$, chúng tôi thấy $g(p)$ đạt cực đại khi $p = \frac{1}{2}$ và khi $p \to 0$ hoặc $p \to 1$ thì $g(p) = p(1-p) \to 0$. Hơn nữa, khi đó chúng tôi có được dãy hàm xấp xỉ:

$$U_t(\boldsymbol{x}) = f(\boldsymbol{x}) + (\frac{a_t}{t} - p)(\frac{g_1(\boldsymbol{x})}{p} - \frac{g_2(\boldsymbol{x})}{1-p})$$

Đặt $h(t,p) = \frac{a_t}{t} - p$, ta có $h(t,p)$ tiến về 0 khi $t \to \infty$. Khi đó:

$$R(g_1, g_2, p) = h(t,p)(\frac{g_1(\boldsymbol{x})}{p} - \frac{g_2(\boldsymbol{x})}{1-p}) \to 0 \text{ khi } t \to \infty$$

Như vậy:

$$U_t(\boldsymbol{x}) = f(\boldsymbol{x}) + R(g_1, g_2, p)$$

trong đó $R(g_1, g_2, p) = h(t,p)(\frac{g_1(\boldsymbol{x})}{p} - \frac{g_2(\boldsymbol{x})}{1-p})$ đóng vai trò là thành phần hiệu chỉnh với $h(t,p) = \frac{a_t}{t} - p$ tiến về 0 khi $t \to \infty$. Như vậy, thành phần hiệu chỉnh $R(g_1, g_2, p)$ phụ thuộc nhiều vào giá trị của tham số Bernoulli $p \in (0,1)$. Do đó, xét về bản chất, tham số Bernoulli $p \in (0,1)$ đóng vai trò là tham số hiệu chỉnh góp phần đưa thuật toán BOPE trở thành một phương pháp hiệu chỉnh hiệu quả để giải bài toán MAP. □

4.2.4. Mở rộng cho bài toán tối ưu không lồi tổng quát

Xét bài toán tối ưu không lồi tổng quát có dạng

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x} \in \Omega} [f(\boldsymbol{x}) = g_1(\boldsymbol{x}) + g_2(\boldsymbol{x})] \tag{4.11}$$

Nhận thấy, BOPE đơn giản nhưng hiệu quả, hơn nữa BOPE được thiết kế không phụ thuộc nhiều vào đặc điểm của hàm mục tiêu $f(\boldsymbol{x})$. Do đó, chúng ta có thể mở rộng BOPE để giải bài toán tối ưu không lồi tổng quát. Chi tiết của thuật toán BOPE tổng quát được trình bày trong Thuật toán 4.2.

Chúng tôi cũng đã làm rõ ưu điểm vượt trội của BOPE so với các thuật toán suy diễn trước đây khi xét trên một số phương diện tiêu chí quan trọng như: Thuật toán có đảm bảo cơ sở lý thuyết cho sự hội tụ hay không? Tốc độ hội tụ là bao nhiêu? Thuộc nhóm thuật toán ngẫu nhiên không? Có khả năng linh hoạt dễ dàng mở rộng áp dụng cho các mô hình bài toán khác hay không? Có khả năng hiệu chỉnh hay không? Chúng tôi cũng tiến hành xem xét BOPE và so

Thuật toán 4.2 BOPE tổng quát giải bài toán tối ưu không lồi

Đầu vào: Tham số Bernoulli $p \in (0, 1)$
Đầu ra: $\boldsymbol{x}^*$ là điểm cực đại của hàm số $f(\boldsymbol{x}) = g_1(\boldsymbol{x}) + g_2(\boldsymbol{x})$ trên miền Ω

Khởi tạo $\boldsymbol{x}_1$ trong miền Ω
$G_1(\boldsymbol{x}) := \frac{g_1(\boldsymbol{x})}{p}$; $G_2(\boldsymbol{x}) := \frac{g_2(\boldsymbol{x})}{1-p}$
$f_1^l := G_1(\boldsymbol{x})$ và $f_1^u := G_2(\boldsymbol{x})$
for $t = 2, 3, \ldots, \infty$ **do**
    Lấy f_t^l có phân phối Bernoulli từ $\{G_1(\boldsymbol{x}), G_2(\boldsymbol{x})\}$ trong đó

$$P(f_t^l = G_1(\boldsymbol{x})) = p; P(f_t^l = G_2(\boldsymbol{x})) = 1 - p$$

    $L_t := \frac{1}{t}\sum_{h=1}^{t} f_h^l$
    $\boldsymbol{a}_t^l := \arg\max_{\boldsymbol{x}\in\Omega} < L_t^{'}(\boldsymbol{x}_t), \boldsymbol{x} >$
    $\boldsymbol{x}_{t+1}^l := \boldsymbol{x}_t + \frac{\boldsymbol{a}_t^l - \boldsymbol{x}_t}{t}$
    Lấy f_t^u có phân phối Bernoulli từ $\{G_1(\boldsymbol{x}), G_2(\boldsymbol{x})\}$ trong đó

$$P(f_t^u = G_1(\boldsymbol{x})) = p; P(f_t^u = G_2(\boldsymbol{x})) = 1 - p$$

    $U_t := \frac{1}{t}\sum_{h=1}^{t} f_h^u$
    $\boldsymbol{a}_t^u := \arg\max_{\boldsymbol{x}\in\Omega} < U_t^{'}(\boldsymbol{x}_t), \boldsymbol{x} >$
    $\boldsymbol{x}_{t+1}^u := \boldsymbol{x}_t + \frac{\boldsymbol{a}_t^u - \boldsymbol{x}_t}{t}$
    $\boldsymbol{x}_{t+1} := \arg\max_{\boldsymbol{x}\in\{\boldsymbol{x}_{t+1}^u,\ \boldsymbol{x}_{t+1}^l\}} f(\boldsymbol{x})$
end for

sánh với các phương pháp suy diễn khác như VB, CVB, CGS, FW, OPE,v.v... Chi tiết kết quả đối chiếu được chúng tôi tổng kết trong Bảng 4.1.

Phương pháp suy diễn	**Tốc độ hội tụ**	**Ngẫu nhiên**	**Linh hoạt**	**Hiệu chỉnh**
VB [39], CVB [40], CVB0 [42]	–	–	–	–
SMM [45], CCCP [44]	–	–	–	–
CGS [43]	–	Có	–	–
PMD [49]	$\mathcal{O}(T^{-1/2})$	Có	–	–
HAMCMC [50]	$\mathcal{O}(T^{-1/3})$	Có	–	–
OPE [28]	$\mathcal{O}(1/T)$	Phân phối đều	Có	–
BOPE	$\mathcal{O}(1/T)$	Phân phối Bernoulli	Có	Có

Bảng 4.1: So sánh về mặt lý thuyết của các phương pháp suy diễn trên các tiêu chuẩn như tốc độ hội tụ, tính ngẫu nhiên, tính linh hoạt và tính hiệu chỉnh. Ký hiệu T là số lần lặp và '-' biểu thị 'không xác định'. Chúng tôi phát hiện BOPE có ưu thế vượt trội so với các phương pháp suy diễn đương đại khác.

4.3. Áp dụng BOPE vào mô hình LDA cho phân tích văn bản

Trong phần này, để làm rõ hiệu quả của BOPE trên khía cạnh thực nghiệm, chúng tôi áp dụng BOPE để giải bài toán MAP trong các mô hình chủ đề. Mô hình chủ đề là một công cụ mạnh mẽ được sử dụng rộng rãi trong khai thác dữ liệu văn bản, do đó chúng tôi sử dụng BOPE là cốt lõi suy diễn từ đó thiết kế các thuật toán học ngẫu nhiên hiệu quả học các mô hình chủ đề.

4.3.1. Suy diễn MAP cho từng văn bản

Chúng tôi tiếp tục xem xét bài toán MAP đối với từng văn bản $\boldsymbol{d}$ trong mô hình chủ đề:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Delta_K} \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^{K} \log \theta_k \tag{4.12}$$

trong đó tham số $\alpha < 1$. Ký hiệu:

$$g_1(\boldsymbol{\theta}) := \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj} \ , \ g_2(\boldsymbol{\theta}) := (\alpha - 1) \sum_{k=1}^{K} \log \theta_k$$

Khi đó bài toán (4.12) đưa về dạng:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Delta_K} [f(\boldsymbol{\theta}) = g_1(\boldsymbol{\theta}) + g_2(\boldsymbol{\theta})] \tag{4.13}$$

Quan sát chúng tôi nhận thấy:

- Thành phần $g_1(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj} < 0$ là log likelihood và $g_2(\boldsymbol{\theta}) = (\alpha - 1) \sum_{k=1}^{K} \log \theta_k > 0$ là log prior của văn bản $\mathbf{d}$.
- Hàm mục tiêu $f(\boldsymbol{\theta}) = g_1(\boldsymbol{\theta}) + g_2(\boldsymbol{\theta})$ bị kẹp giữa hai hàm g_1 và g_2, tức là $g_1(\boldsymbol{\theta}) < f(\boldsymbol{\theta}) < g_2(\boldsymbol{\theta})$.

Khi đó chúng tôi có thể áp dụng BOPE để giải tốt bài toán (4.12). Chi tiết của BOPE để giải bài toán MAP trong mô hình chủ đề LDA được trình bày trong Thuật toán 4.3.

Thuật toán 4.3 BOPE giải bài toán MAP trong mô hình chủ đề LDA

Đầu vào: Văn bản $\mathbf{d}$, tham số Bernoulli $p \in (0, 1)$ và tham số mô hình $\{\boldsymbol{\beta}, \alpha\}$

Đầu ra: $\boldsymbol{\theta}^*$ là điểm cực đại hóa của hàm $f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^{K} \log \theta_k$

Khởi tạo $\boldsymbol{\theta_1}$ trong miền $\overline{\Delta}_K$

$G_1(\theta) := \frac{\sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj}}{p}$; $G_2(\theta) := \frac{(\alpha-1) \sum_{k=1}^{K} \log \theta_k}{1-p}$

$f_1^l := G_1(\theta)$ and $f_1^u := G_2(\theta)$

for $t = 2, 3, \ldots \infty$ **do**

  Lấy f_t^l có phân phối Bernoulli với xác suất p từ tập $\{G_1(\boldsymbol{\theta}),\ G_2(\boldsymbol{\theta})\}$ trong đó

$$P(f_t^l = G_1(\boldsymbol{\theta})) = p \ , \ P(f_t^l = G_2(\boldsymbol{\theta})) = 1 - p$$

  $L_t := \frac{1}{t} \sum_{h=1}^{t} f_h^l$

  $\boldsymbol{e}_t^l := \arg\max_{\boldsymbol{x}\in\overline{\Delta}_K} < L_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} >$

  $\boldsymbol{\theta}_{t+1}^l := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^l - \boldsymbol{\theta}_t}{t}$

  Lấy f_t^u có phân phối Bernoulli với xác suất p từ tập $\{G_1(\boldsymbol{\theta}),\ G_2(\boldsymbol{\theta})\}$ trong đó

$$P(f_t^u = G_1(\boldsymbol{\theta})) = p \ , \ P(f_t^u = G_2(\boldsymbol{\theta})) = 1 - p$$

  $U_t := \frac{1}{t} \sum_{h=1}^{t} f_h^u$

  $\boldsymbol{e}_t^u := \arg\max_{\boldsymbol{x}\in\overline{\Delta}_K} < U_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} >$

  $\boldsymbol{\theta}_{t+1}^u := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}$

  $\boldsymbol{\theta}_{t+1} := \arg\max_{\boldsymbol{\theta}\in\{\boldsymbol{\theta}_{t+1}^l, \boldsymbol{\theta}_{t+1}^u\}} f(\boldsymbol{\theta})$

end for

Chúng tôi tiếp tục chỉ ra trong phần này sự đơn giản của việc sử dụng BOPE để thiết kế thuật toán học nhanh LDA. Cụ thể hơn, dựa trên Online-OPE [28], chúng tôi thiết kế Online-BOPE học LDA từ tập dữ liệu lớn theo cách trực tuyến. Chi tiết về Online-BOPE được trình bày trong Thuật toán 4.4.

Thuật toán 4.4 Online-BOPE học mô hình LDA từ dữ liệu lớn

Đầu vào: Tập dữ liệu huấn luyện $\mathcal{C}$ với D văn bản, $K, \alpha, \eta, \tau > 0, \kappa \in (0.5, 1]$
Đầu ra: $\boldsymbol{\lambda}$

1: Khởi tạo $\boldsymbol{\lambda}^{\mathbf{0}}$ ngẫu nhiên
2: **for** $t = 1, 2, \ldots, \infty$ **do**
3: Lấy mẫu nhỏ $\mathcal{C}_t$ gồm S văn bản.
4: Sử dụng BOPE để suy diễn hậu nghiệm cho mỗi văn bản $\mathbf{d} \in \mathcal{C}_t$, cho bởi biến toàn cục $\boldsymbol{\beta}^{t-1} \propto \boldsymbol{\lambda}^{t-1}$ trước đó để nhận được chủ đề hỗn hợp $\boldsymbol{\theta}_d$. Tính toán $\boldsymbol{\phi}_d$ theo

$$\phi_{djk} \propto \theta_{dk}\beta_{kj}$$

5: Với mỗi $k \in \{1, 2, .., K\}$, biến toàn cục $\hat{\boldsymbol{\lambda}}_k$ cho $\mathcal{C}_t$ bởi

$$\hat{\lambda}_{kj} = \eta + \frac{D}{S}\sum_{\mathbf{d}\in\mathcal{C}_t} d_j\phi_{djk}$$

6: Cập nhật biến toàn cục

$$\boldsymbol{\lambda}^t := (1-\rho_t)\boldsymbol{\lambda}^{t-1} + \rho_t\hat{\boldsymbol{\lambda}}$$

trong đó $\rho_t = (t+\tau)^{-\kappa}$
7: **end for**

Thuật toán Online-BOPE sử dụng BOPE để thực hiện suy diễn MAP cho các văn bản riêng lẻ và lược đồ trực tuyến [9, 32, 99] để suy diễn các biến toàn cục (chủ đề). Do đó, bản chất ngẫu nhiên xuất hiện trong cả hai giai đoạn suy diễn cục bộ và toàn cục. Suy diễn MAP về các biến cục bộ của BOPE có đảm bảo lý thuyết về tốc độ hội tụ nhanh, giúp thuật toán học Online-BOPE tốt hơn các thuật toán học hiện có. Chúng tôi tiến hành thực nghiệm với thuật toán học Online-BOPE trên năm bộ dữ liệu lớn và so sánh kết quả của Online-BOPE với các phương pháp học khác như Online-VB, Online-CVB0, Online-CGS và Online-OPE.

4.3.2. Đánh giá thực nghiệm

Phần này được dành cho việc điều tra các hành vi thực tế của BOPE và sự hữu ích khi sử dụng BOPE để thiết kế thuật toán học ngẫu nhiên mới học các mô hình chủ đề ở quy mô lớn.

Các thuật toán suy diễn: Chúng tôi tiến hành so sánh thuật toán suy diễn BOPE với các phương pháp suy diễn đương đại như: Variational Bayes (VB) [39], Collapsed variational Bayes (CVB0) [42], Collapsed Gibbs sampling (CGS) [43], Online maximum a Posterior Estimation (OPE)[28]. Một số nghiên cứu trước đây như [42, 43, 109] đã chỉ ra CVB0 và CGS có hiệu quả tốt cho bài toán suy diễn. Do đó, chúng có thể được coi là phương pháp suy diễn tiên tiến nhất.

Bộ dữ liệu	Kích thước bộ dữ liệu	Độ dài văn bản TB	Từ điển V
New York Times	300,000	325.13	102,661
PubMed	330,000	65.12	141,044
Yahoo	517,770	4.73	24,420
Twitter	1,457,687	10.14	89,474
NYT-Titles	1,664,127	5.15	55,488

Bảng 4.2: Bảng mô tả năm bộ dữ liệu thực nghiệm

Các phương pháp học: Chúng tôi tiến hành các thực nghiệm để điều tra tính hiệu quả của Online-BOPE khi so sánh với các phương pháp học ngẫu nhiên khác như: Online-CGS [43], Online-CVB0 [100], Online-VB [32], Online-OPE [28]. Online-CGS là một thuật toán lai, trong đó CGS được sử dụng để ước tính phân phối biến cục bộ ($\mathbf{z}$) trong tài liệu và VB được sử dụng để ước tính phân phối biến toàn cục ($\boldsymbol{\lambda}$). Online-CVB0 là phiên bản trực tuyến của thuật toán "batch" trong [42], trong đó suy diễn cho một văn bản được thực hiện bởi CVB0 còn Online-VB là một thuật toán học ngẫu nhiên mà suy diễn cho một văn bản được thực hiện bởi thuật toán VB.

a. Các bộ dữ liệu thực nghiệm

Chúng tôi tiếp tục xem xét hiệu quả của thuật toán BOPE về mặt thực nghiệm. Trong các thực nghiệm, chúng tôi sử dụng năm bộ dữ liệu văn bản lớn thuộc 2 nhóm: dữ liệu văn bản dài (bao gồm các văn bản như các bài báo có độ dài văn bản không quá ngắn thường từ vài chục đến vài trăm từ trên một văn bản) và dữ liệu văn bản ngắn (bao gồm các văn bản có độ dài văn bản rất ngắn, trung bình chỉ khoảng 5-10 từ trên một văn bản). Mô tả chi tiết cho từng tập dữ liệu được hiển thị trong Bảng 4.2.

- Dữ liệu văn bản dài: Chúng tôi sử dụng hai bộ dữ liệu văn bản dài: PubMed bao gồm 330.000 bài viết liên quan về sức khỏe từ PubMed Central và New York Times (NYT) bao gồm 300.000 bài tin tức của thời báo New York Times[1]. Các văn bản của các bộ dữ liệu này là các bài báo nên độ dài của các văn bản thường lớn (từ vài chục đến vài trăm từ trên một văn bản).

- Dữ liệu văn bản ngắn: Chúng tôi sử dụng ba bộ dữ liệu lớn gồm các văn bản ngắn để đánh giá [110]: Yahoo question bao gồm các câu hỏi từ mạng hỏi đáp của Yahoo; Twitter Tweets bao gồm các bài tweets từ mạng xã hội twitter; NYT-Titles bao gồm các tiêu đề của các bài báo trong tờ thời báo New York Times[2]. Các bộ dữ liệu này được tiền xử lý bằng cách mã hóa, loại bỏ các từ dừng, loại bỏ các từ có tần số thấp (xuất hiện trong ít hơn 3 văn bản) và xóa các văn bản cực ngắn (dưới 3 từ).

[1]Hai bộ dữ liệu này được lấy từ nguồn http://archive.ics.uci.edu/ml/datasets
[2]Ba bộ dữ liệu ngắn được lấy từ nguồn: answer.yahoo.com, twitter.com và www.nytimes.com

Các văn bản ngắn đặt ra những khó khăn khác nhau đối với các bài toán phân tích văn bản [110, 111, 112]. Do đó, việc sử dụng cả văn bản dài và văn bản ngắn trong cuộc điều tra của chúng tôi sẽ cho thấy rõ hơn hiệu suất của các phương pháp mới đề xuất so với các phương pháp đối sánh. Đối với mỗi tập dữ liệu, chúng tôi lấy ngẫu nhiên 1000 văn bản để kiểm tra và sử dụng phần còn lại để huấn luyện.

b. Thiết lập tham số

- *Tham số mô hình*: Chúng tôi thiết lập số chủ đề $K = 100$, siêu tham số $\alpha = \frac{1}{K}$ và tham số Dirichlet $\eta = \frac{1}{K}$. Các tham số này thường được sử dụng trong các mô hình chủ đề. Sự lựa chọn (α, η) như vậy đã được quan sát là hoạt động tốt trong nhiều nghiên cứu trước đây [92, 32, 100].

- *Tham số suy diễn*: Chúng tôi chọn tham số Bernoulli $p \in \{0.1, 0.2, .., 0.8, 0.9\}$ và số lần lặp của thuật toán suy diễn là $T = 50$. Nhiều nhất 50 lần lặp để BOPE, OPE và VB thực hiện suy diễn. Chúng tôi sẽ dừng thuật toán VB nếu sự cải thiện của giới hạn dưới không tốt hơn 10^{-4}. 50 mẫu đã được sử dụng trong CGS trong đó 25 mẫu đầu tiên bị loại bỏ và số còn lại được sử dụng để xấp xỉ phân phối hậu nghiệm. 50 lần lặp được sử dụng để suy diễn trong CVB0. Số lượng mẫu/lần lặp đó thường đủ để thu được kết quả suy diễn tốt [43, 100].

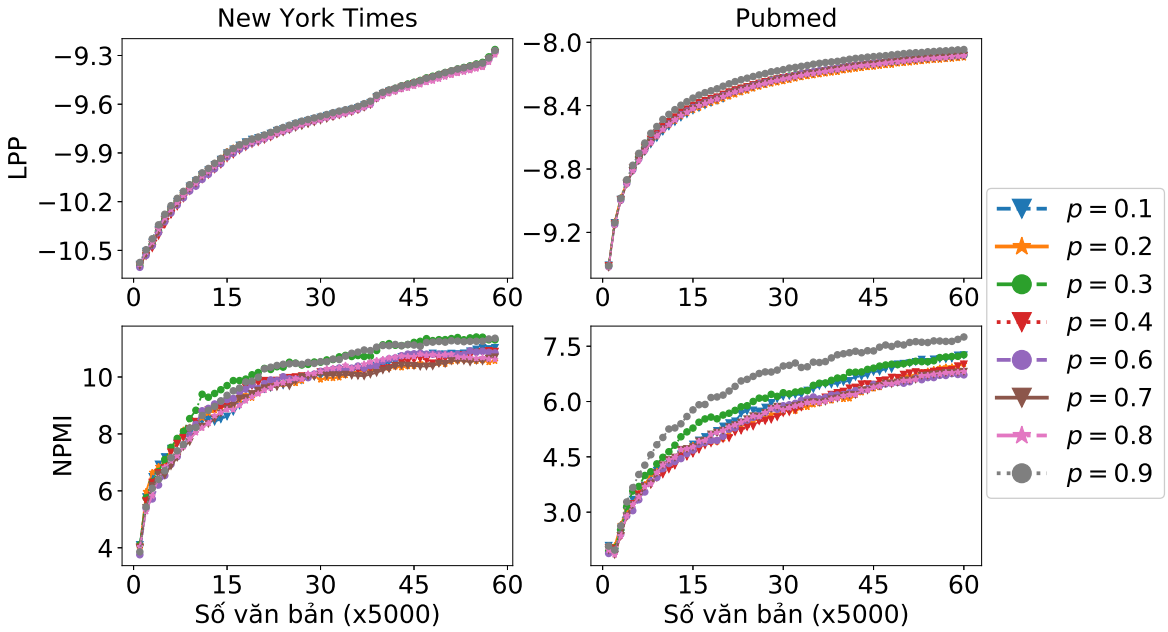
- *Tham số học*: Chúng tôi thiết lập kích thước mini-batch $S = |C_t| = 5000$, $\kappa = 0,9$, $\tau = 1$. Sự lựa chọn các tham số học này đã được tìm thấy dẫn đến hiệu suất cạnh tranh của Online-VB [32] và Online-CVB0 [100]. Do đó, nó đã được sử dụng trong thực nghiệm của chúng tôi để tránh sự thiên vị có thể xảy ra. Chúng tôi đã sử dụng các giá trị mặc định cho một số tham số khác trong Online-CVB0.

c. Độ đo đánh giá thực nghiệm

Chúng tôi tiếp tục sử dụng hai độ đo *Log Predictive Probability (LPP)* [43] và *Normalised Pointwise Mutual Information (NPMI)* [102]. Nếu LPP đo lường tính dự đoán và khái quát hóa của một mô hình đối với dữ liệu mới thì NPMI đánh giá chất lượng ngữ nghĩa của một chủ đề riêng lẻ. Độ đo LPP và NPMI càng cao càng tốt thể hiện phương pháp học càng hiệu quả. Chi tiết về cách tính các độ đo này được trình bày trong Phụ lục A và B.

d. Kết quả thực nghiệm

Theo hiểu biết của chúng tôi, thuật toán suy diễn đóng vai trò quan trọng trong các thuật toán học LDA. Do vậy đánh giá hiệu quả của thuật toán suy



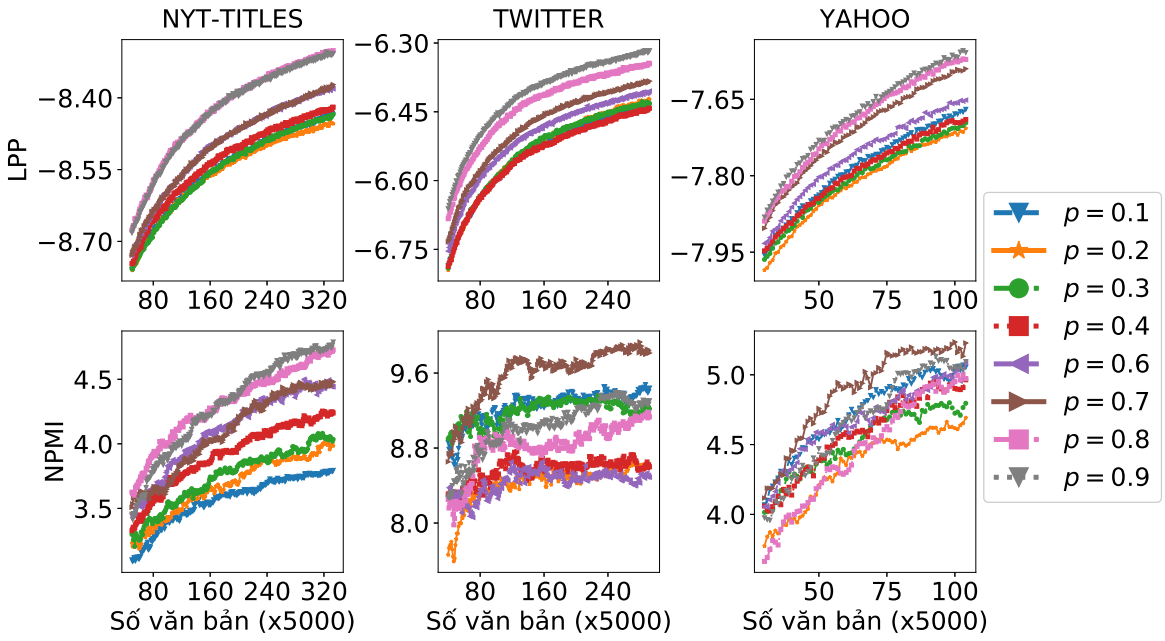
Hình 4.1: Kết quả của Online-BOPE với giá trị tham số Bernoulli p khác nhau trên bộ dữ liệu New York Times và PubMed với độ đo LPP và NPMI. Độ đo càng cao thể hiện mô hình càng tốt.

diễn BOPE sẽ được xem xét gián tiếp thông qua hiệu quả của thuật toán học Online-BOPE.

Trước tiên, chúng tôi tiến hành xem xét vai trò của tham số Bernoulli p trong BOPE. Chúng tôi chọn giá trị của p rời rạc trong tập $\{0.1, 0.2, .., 0.9\}$ và thực hiện thuật toán học Online-BOPE trên năm bộ dữ liệu. Chúng tôi phát hiện ra rằng p ảnh hưởng rất nhiều đến hiệu suất của thuật toán học (hay bản chất là thuật toán suy diễn) trên cả hai loại dữ liệu là văn bản dài và văn bản ngắn. Kết quả của Online-BOPE trên bộ dữ liệu New York Times và PubMed được chỉ ra trong Hình 4.1.

Theo Hình 4.1, chúng tôi thấy rằng hai độ đo LPP và NPMI của thuật toán học Online-BOPE thay đổi theo giá trị của tham số Bernoulli p, điều đó phản ánh tham số Bernoulli p có tác động đến hiệu quả của BOPE. Đặc biệt, sự khác biệt đó được thấy rõ hơn trên trên độ đo NPMI và trên bộ PubMed. Có thể lý giải sự khác biệt trên PubMed rõ hơn New York Times là do ảnh hưởng của độ dài văn bản trong mỗi tập dữ liệu. Mặc dù New York Times và PubMed đều là các bộ dữ liệu văn bản dài, tuy nhiên bộ PubMed có độ dài trung bình của các văn bản chỉ khoảng 65 từ/văn bản trong khi đó độ dài trung bình của các văn bản trong New York Times khoảng 325 từ/văn bản. Như vậy bộ PubMed có độ dài văn bản ngắn hơn bộ New York Times khá nhiều. Như vậy thay đổi tham số Bernoulli p góp phần điều chỉnh tỷ lệ thành phần likelihood và prior trong hàm mục tiêu của bài toán MAP cần tối ưu.

Theo các nghiên cứu trước đây [28, 113], mô hình LDA thường không làm

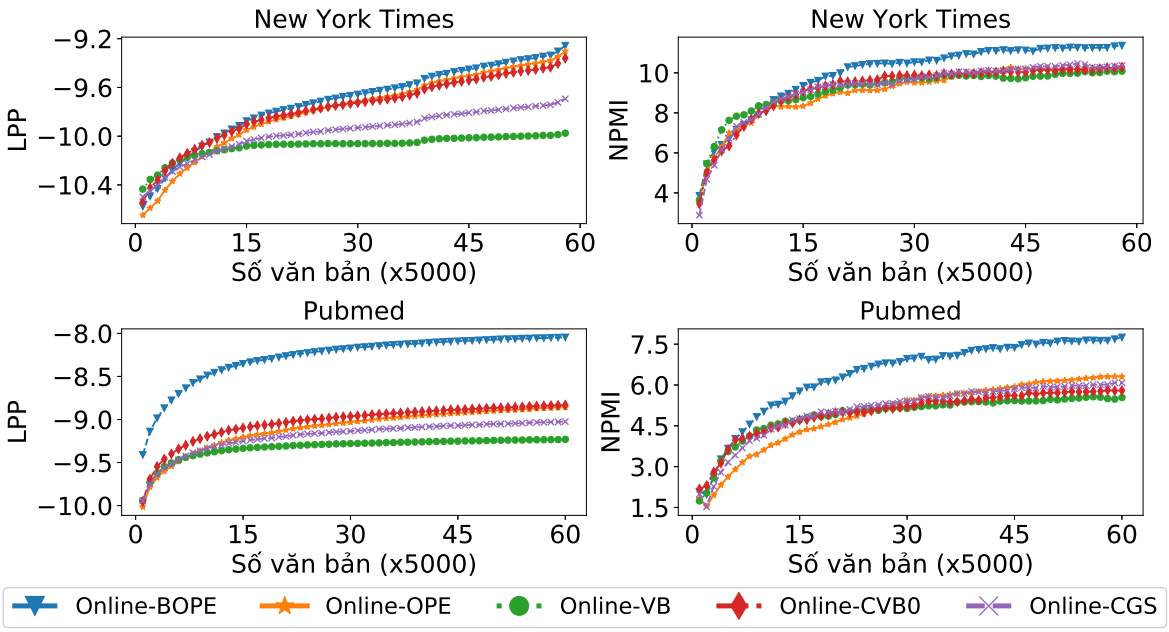


Hình 4.2: Kết quả của Online-BOPE với giá trị tham số Bernoulli p khác nhau trên độ đo LPP và NPMI và trên các bộ dữ liệu văn bản ngắn. Độ đo càng cao càng tốt.

việc tốt trên các dữ liệu ngắn, thậm chí xảy ra hiện tượng quá khớp. Với cách thiết kế của BOPE, chúng tôi chỉ ra rằng Online-BOPE có thể giúp LDA làm việc tốt hơn trên các dữ liệu NYT-Titles, Twitter tweets và Yahoo question. Đây là ba bộ dữ liệu có trung bình độ dài văn bản ngắn, chỉ khoảng 4-10 từ/văn bản. Chi tiết kết quả được trình bày trong Hình 4.2.

Thông qua Hình 4.2, chúng tôi cho thấy rằng tham số Bernoulli p có tác động lớn đến hiệu quả của Online-BOPE, đặc biệt là trên bộ dữ liệu văn bản ngắn. Thực nghiệm trên bộ dữ liệu NYT-Titles, Twitter và Yahoo question, chúng tôi thấy khi chọn p có xu hướng gần 1, chẳng hạn như $p = 0.9$ trong thực nghiệm thì Online-BOPE thường cho kết quả cao hơn. Đây là một gợi ý để chúng ta chọn tham số Bernoulli p phù hợp cho BOPE. Đồng thời, điều này hoàn toàn phù hợp với cơ sở lý thuyết của thuật toán BOPE đã được chứng minh ở trên. Đó là khi làm việc với các bộ dữ liệu có độ dài văn bản ngắn, chúng ta cần phải khai thác triệt để thành phần likelihood (trong trường hợp này các văn bản ngắn nên thành phần likelihood không chứa nhiều thông tin về của dữ liệu quan sát), đồng thời p lớn giúp cho thành phần hiệu chỉnh đủ nhỏ giúp cho thuật toán hiệu quả và tránh được hiện tượng quá khớp.

LDA có thể phù hợp với các bộ dữ liệu văn bản lớn bằng cách sử dụng tối ưu hóa ngẫu nhiên [114, 32]. Tuy nhiên, LDA có thể thất bại khi đối mặt với các bộ dữ liệu có lượng từ điển lớn nhưng độ dài văn bản ngắn. Chúng tôi đánh giá hiệu quả của BOPE ước lượng MAP trong các mô hình chủ đề thông qua Online-BOPE để học LDA sử dụng hai độ đo LPP và NPMI. Hơn nữa, chúng tôi so



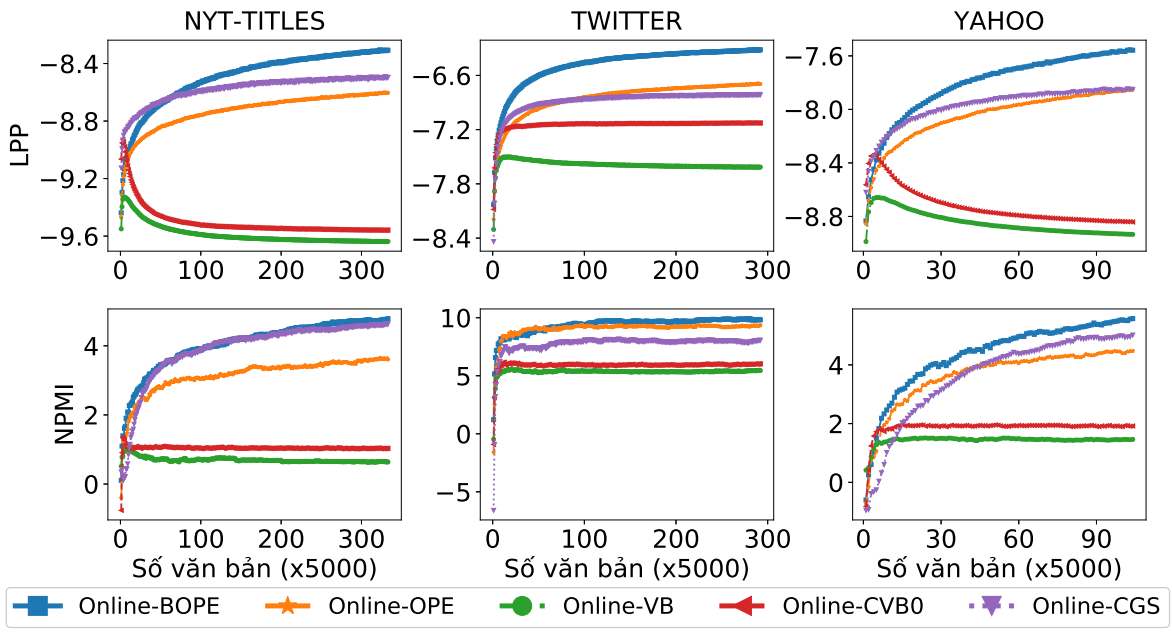
Hình 4.3: Kết quả của các phương pháp học ngẫu nhiên trên New York Times và PubMed. Độ đo cao hơn thì tốt hơn. Chúng tôi nhận thấy Online-BOPE thường cho kết quả tốt nhất.

sánh Online-BOPE với các thuật toán học khác như Online-VB, Online-CVB0, Online-CGS và Online-OPE. Tất cả các thực nghiệm được tiến hành trên cả hai loại dữ liệu văn bản ngắn và văn bản dài.

Với dữ liệu văn bản dài: Chúng tôi so sánh Online-BOPE với Online-VB, Online-CVB0, Online-CGS và Online-OPE trên hai bộ dữ liệu New York Times và PubMed. Kết quả chi tiết được mô tả trong Hình 4.3.

Theo Hình 4.3, chúng tôi thấy rằng độ đo LPP và NPMI của các phương pháp học đều tăng theo số lượng văn bản đã học. Lý giải cho vấn đề này chính là mô hình LDA thường sử dụng giả thiết về sự đồng xuất hiện của hai từ trong cùng một văn bản, do đó LDA phù hợp với dữ liệu thông thường như New York Times và PubMed. Tuy nhiên, có sự khác biệt rất lớn về độ đo LPP và NPMI của Online-BOPE so với các phương pháp học khác như Online-VB, Online-CVB0, Online-CGS và Online-OPE, nhất là trên bộ PubMed. Điều này giải thích rằng với tham số Bernoulli p phù hợp thì BOPE đặc biệt phù hợp với các bộ dữ liệu với văn bản không quá dài tức là có thành phần likelihood và prior không quá chênh lệch. Bởi vì tham số Bernoulli p kiểm soát và điều chỉnh tỷ lệ likelihood và prior cho phù hợp trong ước lượng MAP.

Với dữ liệu văn bản ngắn: Chúng tôi tiếp tục điều tra tính hiệu quả của Online-BOPE trên tập các văn bản ngắn như Twitter, NYT-Titles, Yahoo (xem Hình 4.4). Chúng tôi cho thấy rằng BOPE giúp Online-BOPE tốt hơn các phương pháp so sánh trên các văn bản ngắn ở một số khía cạnh như tính dự đoán, tính tổng quát và ngăn chặn sự quá khớp.

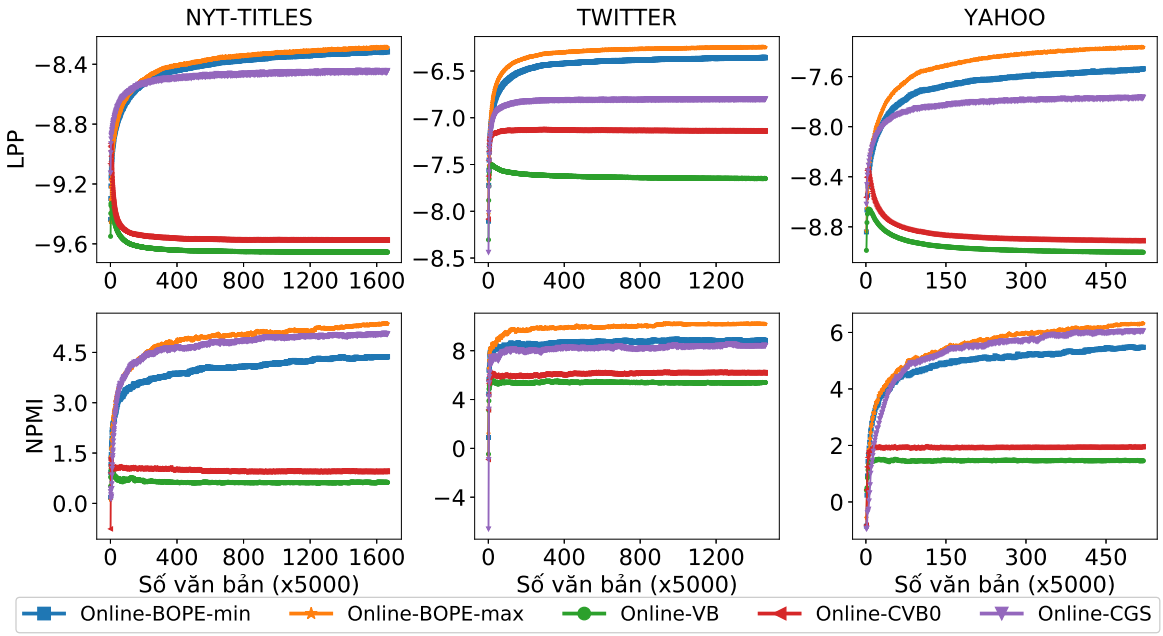


Hình 4.4: Kết quả của các phương pháp học ngẫu nhiên trên các bộ dữ liệu văn bản ngắn: NYT-Titles, Twitter và Yahoo. Chúng tôi thấy Online-BOPE thường cho kết quả tốt nhất trên cả hai độ đo LPP và NPMI.

Chúng tôi quan sát thấy sự quá khớp của Online-VB và Online-CVB0 trong Hình 4.4. Cụ thể chúng tôi thấy độ đo LPP và NPMI của Online-VB và Online-CVB0 bị giảm theo số lượng văn bản học trong khi độ đo LPP và NPMI của Online-CGS, Online-OPE và Online-BOPE vẫn luôn tăng theo số lượng văn bản học được. Điều đó có nghĩa là khả năng tổng quát của mô hình giảm khi học bởi Online-VB và Online-CVB và trên ba bộ dữ liệu văn bản ngắn, đặc biệt là NYT-Titles và Yahoo.

Chúng tôi tiến hành huấn luyện mô hình trên mỗi bộ dữ liệu 5 lần (5 epochs) và ghi lại kết quả thực hiện thuật toán học trong Hình 4.5. Đối với mỗi tập dữ liệu, chúng tôi đã thực hiện Online-BOPE với tham số Bernoulli $p \in \{0.1, 0.2, .., 0.9\}$ sau đó ghi lại kết quả tốt nhất (ký hiệu là Online-BOPE-max) và kết quả tồi nhất (ký hiệu là Online-BOPE-min) và so sánh với Online-VB, Online-CVB và Online-CGS.

Chúng tôi phát hiện được chất lượng của Online-BOPE vẫn tốt sau 5 epochs. Tuy nhiên, hiện tượng quá khớp của Online-VB và Online-CVB0 xảy ra càng nhiều. Độ đo LPP và NPMI của Online-VB và Online-CVB0 có xu hướng giảm mạnh theo số văn bản huấn luyện, nhất là độ đo LPP, tức là khả năng tổng quát của mô hình giảm dần theo số văn bản học và số epochs. Từ Hình 4.5, chúng tôi cũng thấy hiện tượng over-fitting xảy ra nhất trên bộ NYT-Titles và Yahoo. Đây là hai bộ dữ liệu có độ dài văn bản rất ngắn, trung bình chỉ khoảng 4 đến 5 từ trên một văn bản). Chúng tôi có thể lý giải kết quả trong Hình 4.4 và Hình 4.5 như sau:



Hình 4.5: Kết quả của các phương pháp học ngẫu nhiên trên các dữ liệu văn bản ngắn: NYT-Titles, Twitter và Yahoo sau 5 epochs. Chúng tôi phát hiện ra rằng Online-BOPE cho kết quả tốt nhất.

- Nhóm Online-VB và Online-CVB0 có bước suy diễn là tất định, mặc dù thuật toán suy diễn CVB0 có một chút xấp xỉ "nhiễu" trong công thức cập nhật N_{dk}^{θ}, tuy nhiên mục đích chính của thuật toán suy diễn này đó là tìm ra ϕ_{dn}^{k} theo công thức đã cho;
- Nhóm Online-CGS, Online-OPE và Online-BOPE có bước suy diễn tương ứng là CGS, OPE và BOPE tiếp cận theo hướng lấy mẫu ngẫu nhiên hoặc sử dụng phân phối ngẫu nhiên.

Như vậy, nhóm thuật toán suy diễn ngẫu nhiên cho kết quả vượt trội hơn hẳn so với nhóm suy diễn tất định, đồng thời không gặp hiện tượng quá khớp. Hơn nữa, Online-BOPE cho kết quả cao hơn Online-CGS và Online-OPE. Đạt được điều đó là do BOPE đã kế thừa những đặc tính tốt của OPE như đảm bảo lý thuyết về sự hội tụ nhanh, sự ổn định, không làm thay đổi biến toàn cục khi thực hiện suy diễn với từng văn bản riêng lẻ. Phân phối Bernoulli với tham số $p \in (0, 1)$ và hai biên ngẫu nhiên giúp thuật toán có tính tổng quát tốt, tính tương thích với các bộ dữ liệu và mô hình cao hơn, đồng thời góp phần đẩy nhanh tốc độ hội tụ. Một ưu điểm của BOPE chính là sự đóng góp của phân phối Bernoulli với tham số $p \in (0, 1)$. Nó đóng vai trò là tham số hiệu chỉnh giúp BOPE có vai trò của một kỹ thuật hiệu chỉnh giúp thoát khỏi hiện tượng quá khớp hiệu quả nhất.

4.4. Áp dụng BOPE cho bài toán hệ gợi ý

Chúng tôi đã thành công khi ứng dụng BOPE để giải bài toán MAP trong phân tích văn bản, đặc biệt khi phải đối mặt với các bộ dữ liệu văn bản ngắn. Với mục đích làm rõ hơn hiệu quả của BOPE khi có thể áp dụng trên nhiều mô hình, chúng tôi tiếp tục điều tra BOPE khi ứng dụng giải bài toán MAP trong mô hình CTMP (viết tắt của Collaborative Topic Model for Poisson distributed ratings model) [115]. Biết rằng CTMP là một mô hình lai có nhiều ưu thế vượt trội trong bài toán hệ gợi ý.

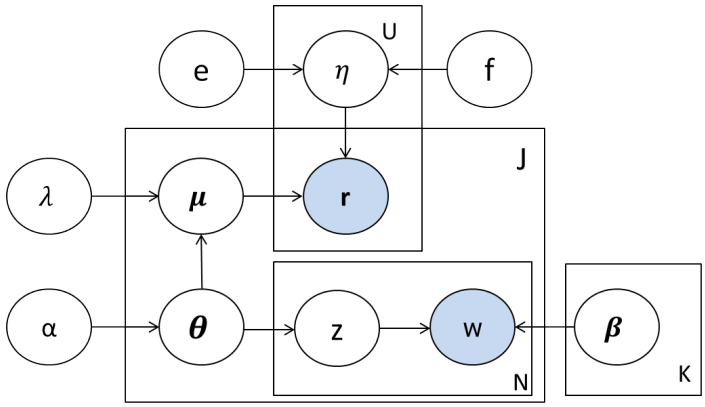
4.4.1. Mô hình CTMP

Hệ thống gợi ý (recommender system) là một hệ thống có khả năng gợi ý các sản phẩm (items) mà nó cho rằng phù hợp đối với người dùng (user) dựa trên một số tiêu chí nào đó. Ví dụ, dựa trên lịch sử giao dịch người dùng hoặc những người dùng có những hành vi tương tự, hệ gợi ý sẽ tính toán và đưa ra dự đoán về độ phù hợp của sản phẩm nào đó và dựa trên những kết quả đó đưa ra những gợi ý cho người dùng. Các thành phần cơ bản của một bài toán gợi ý bao gồm:

- Một lượng hữu hạn U người dùng và I sản phẩm.
- Các thông tin về người dùng: hồ sơ cá nhân, tuổi, giới tính, sở thích,...
- Các thông tin về sản phẩm: tên sản phẩm, mô tả, danh mục, thể loại,...
- Ma trận tương tác R giữa người dùng và sản phẩm, các tương tác ở đây là các hành động của người dùng đối với sản phẩm, ví dụ như: xem, chọn, mua, hoặc đánh giá sản phẩm,...

Hai hướng tiếp cận truyền thống để xây dựng hệ gợi ý: Lọc dựa trên nội dung (Content-based filtering) và lọc cộng tác (Collaborative filtering). CTMP [115] là một mô hình gợi ý xây dựng dựa trên ý tưởng kết hợp sử dụng nội dung mô tả sản phẩm trước khi đưa vào phân rã Poisson. Việc thông tin mô tả sản phẩm của mô hình CTMP được khai thác thông qua mô hình LDA và tận dụng những ưu điểm của phương pháp phân rã Poisson phân cấp. Chúng tôi sẽ sử dụng các ký hiệu sau trong phần này:

- U, J: Số lượng người dùng và sản phẩm tương ứng trong bộ dữ liệu.
- $w_j = \{c_j^\nu\}_{\nu=1}^V$: Biểu diễn "bag-of-word" cho item j trong đó c_j^ν biểu thị tần suất của từ ν trong nội dung/mô tả của mục j.
- V: Kích thước từ điển của nội dung sản phẩm.
- $D = \{r_{uj}, w_j\}_{u=1,j=1}^{U,J}$: bộ dữ liệu xếp hạng ngầm định r_{uj} và nội dung mục (w_j). Xếp hạng được biểu thị bằng ma trận $R = \{r_{uj}\}_{U \times J}$, cho biết xếp hạng



Hình 4.6: Mô hình Collaborative Topic Model for Poisson distributed ratings (CTMP).

mà người dùng u đã đưa ra cho mục j. Mỗi xếp hạng r_{uj} có thể có giá trị 1 (cho biết rằng người dùng u thích sản phẩm j) hoặc 0 (chỉ ra rằng người dùng u không thích điều đó hoặc đơn giản là không biết về mặt hàng j).

- K: Số chủ đề
- $\boldsymbol{\beta} = \{\beta_{k\nu}\}_{K\times V}$: Biểu diễn chủ đề. Mỗi chủ đề k là một phân phối trên bộ từ điển, và nó được biểu diễn bởi véc tơ $\beta_k = \{\beta_{k\nu}\}_{V\times 1}$ ($\sum_{\nu=1}^{V} \beta_{k\nu} = 1, \beta_{k\nu} \geq 0$)
- $\theta_{1:j}$: Biểu diễn nội dung sản phẩm ở mức chủ đề. Véc tơ $\theta_j = \{\theta_{jk}\}_{K\times 1}$ là một phân phối trên các chủ đề ($\sum_{k=1}^{K} \theta_{jk} = 1, \theta_{jk} \geq 0$), là một biểu diễn khác của nội dung sản phẩm trong không gian chủ đề.

Các hệ thống đề xuất thường sử dụng phản hồi từ người dùng để đưa ra gợi ý về các sản phẩm. Các phản hồi có thể được cung cấp rõ ràng hoặc ngầm định bởi người dùng. Một số hệ thống cũng sử dụng nội dung văn bản, chẳng hạn như mô tả sản phẩm hoặc nội dung tin tức, để hiểu thêm về sở thích của người dùng và sau đó đưa ra gợi ý chính xác. Các hệ thống dựa trên bộ lọc cộng tác chủ yếu chỉ sử dụng phản hồi của người dùng, trong khi đó một hệ thống kết hợp có thể sử dụng cả phản hồi và nội dung. Nhóm tác giả trong [115], đã xây dựng mô hình CTMP (Collaborative Topic Model for Poisson) dựa trên ý tưởng kết hợp hai mô hình CTR [116] và CTPF [117] nhằm tận dụng những ưu điểm của cả hai mô hình và hạn chế tối đa những nhược điểm của từng mô hình. CTMP là sự kết hợp giữa kỹ thuật mô hình hóa chủ đề (sử dụng LDA) để sinh nội dung cho mỗi sản phẩm và phân tích Poisson để biểu diễn cho dữ liệu rating rời rạc. Chúng ta có thể dễ dàng nhận thấy việc sử dụng mô hình hóa lấy ý tưởng từ mô hình CTR và phân tích Poisson lấy ý tưởng từ mô hình CTPF. Mô hình sinh của CTMP cụ thể như sau:

Mô hình sinh của CTMP cụ thể như sau:

1. Với mỗi người dùng u, lấy η_u trong đó $\eta_{uk} \sim \text{Gamma}(e, f)$

2. Với mỗi item j:

   (a) Lấy véc tơ tỷ lệ chủ đề $\boldsymbol{\theta}_j \sim \text{Dirichlet}(\alpha)$

   (b) Với từ thứ n của sản phẩm j:

      i. Lấy chỉ số chủ đề $z_{jn} \sim \text{Categorical}(\boldsymbol{\theta}_j)$

      ii. Lấy $w_{jn} \sim \text{Categorical}(\beta_{z_{jn}})$

   (c) Lấy nhân tố ẩn $\mu_j \sim \mathcal{N}(\boldsymbol{\theta}_j, \boldsymbol{\lambda}^{-1}\mathbb{I}_K)$

3. Với mỗi cặp user-item (u, j), lấy $r_{uj} \sim \text{Poisson}(\eta_u^T \mu_j)$

Học CTMP: Chúng ta đã biết tính toán chính xác phân phối hậu nghiệm của các biến ẩn:

$$P(\boldsymbol{\theta}, \mu, \eta | D, \alpha, \boldsymbol{\beta}, \boldsymbol{\lambda}, e, f) = \frac{P(\boldsymbol{\theta}, \mu, \eta, D | \alpha, \boldsymbol{\beta}, \boldsymbol{\lambda}, e, f)}{P(D | \alpha, \beta, \boldsymbol{\lambda}, e, f)} \tag{4.14}$$

là không khả thi, nên suy diễn chính xác là không thể. Có hai cách tiếp cận chính: ước lượng điểm bằng cách ước lượng tối đa xác suất hậu nghiệm (MAP) hoặc học Bayes đầy đủ bằng các phương pháp gần đúng như lấy mẫu MCMC và phương pháp biến phân [75]. Trong quá trình học mô hình CTMP, chúng ta phải cập nhật véc tơ tỷ lệ chủ đề θ_j. Theo [115], tính ước lượng điểm của tỷ lệ chủ đề địa phương θ_j từ hàm mục tiêu:

$$g(\boldsymbol{\theta}_j) = (\alpha - 1) \sum_k \log \theta_{jk} + \sum_\nu c_j^\nu \log \left(\sum_k \theta_{jk} \beta_{k\nu} \right) - \frac{\lambda}{2} \|\theta_j - \mu_j\|_2^2 \tag{4.15}$$

Chúng tôi phát hiện ra rằng hàm mục tiêu $g(\theta_j)$ không phải là lồi khi $\alpha < 1$. Trong [115], các tác giả đã sử dụng OPE [28] để tìm tối ưu θ_j. Lưu ý rằng OPE là một thuật toán tối ưu hóa lặp, là biến thể ngẫu nhiên của thuật toán Frank-Wolfe. OPE cung cấp lợi thế đáng kể cho tính toán, với tốc độ hội tụ nhanh $\mathcal{O}(1/T)$ và chất lượng đã được chứng minh. Ký hiệu:

$$g_1 = (\alpha - 1) \sum_k \log \theta_{jk} + \sum_\nu c_j^\nu \log \left(\sum_k \theta_{jk} \beta_{k\nu} \right) \ , \ g_2 = -\frac{\lambda}{2} \|\theta_j - \mu_j\|_2^2$$

khi đó hàm mục tiêu $g(\theta_j)$ trong (4.15) được viết lại dưới dạng $g = g_1 + g_2$.

Như đã đề cập ở trên, chúng tôi nhận thấy BOPE có nhiều ưu thế vượt trội hơn OPE. Vì vậy, chúng tôi có thể áp dụng BOPE để học θ_j trong mô hình CTMP thay thế OPE đã thực hiện trong [115]. Chi tiết của thuật toán học CTMP được trình bày trong Thuật toán 4.5.

4.4.2. Đánh giá thực nghiệm

Nhóm tác giả trong [115] đã sử dụng thuật toán OPE để học tham số θ_j trong mô hình CTMP. Chúng tôi đã chứng minh được thuật toán BOPE cải

Thuật toán 4.5 Học CTMP bằng phương pháp Coordinate ascent

Đầu vào: Dữ liệu quan sát w, r, tham số Bernoulli $p \in (0,1)$ và siêu tham sốα, $\boldsymbol{\lambda}$, e và f
Đầu ra: Ước lượng $\boldsymbol{\theta}, \mu, \phi_{uj}, shp_{uk}, rte_{uk}$ and $\boldsymbol{\beta}$

Khởi tạo: Khởi tạo $\boldsymbol{\theta}, \boldsymbol{\beta}$ bằng các ước lượng tương ứng từ LDA [39].
repeat
  for $j = 1 : J$ **do**
    Cập nhật θ_j bằng thuật toán BOPE
    Cập nhật μ_j theo như bài báo [115]
  end for
  for u=1:U, k=1:K **do**
    Cập nhật tham số biến phân theo như Bảng 2 trong bài báo [115].
    $\phi_{ujk} \propto \exp[\log \mu_{jk} + \psi(shp_{uk}) - \log(rte_{uk})]$ $\forall j$ nếu $r_{uj} > 0$
    $shp_{uk} \leftarrow e + \sum_j r_{uj}\phi_{uj}$
    $rte_{uk} \leftarrow f + \sum_j r_{uj}$
    $\beta_{k\nu} \propto \sum_j c_j^{\nu}\theta_{jk}$, $\forall k, \nu$
  end for
until hội tụ.

tiến hiệu quả hơn OPE trên nhiều phương diện như tính linh hoạt, có khả năng tổng quát hóa và hiệu chỉnh cao. Do đó trong phần này, chúng tôi đánh giá hiệu quả thực nghiệm của việc sử dụng thuật toán BOPE để học tham số θ_j trong mô hình CTMP (gọi là CTMP-BOPE) thay thế cho thuật toán OPE trong mô hình CTMP ban đầu (gọi là CTMP-OPE). Chúng tôi tiến hành đánh giá thực nghiệm trên bộ dữ liệu CiteUlike và MovieLens 1M và đánh giá hiệu quả mô hình thông qua hai độ đo: độ chính xác (Precision) và độ bao phủ (Recall) khi so sánh với các mô hình khác. Chúng tôi thiết lập tham số tiên nghiệm Gamma $e = f = 0,3$ trong tất cả các thực nghiệm.

a. Các bộ dữ liệu thực nghiệm

Chứng minh sự hiệu quả của thuật toán BOPE khi áp dụng vào suy diễn tham số $\boldsymbol{\theta}_j$ trong mô hình CTMP, chúng tôi tiến hành thực nghiệm mô hình CTMP trên 2 bộ dữ liệu CiteULike[3] và bộ dữ liệu Movielens 1M[4]. Chi tiết hai bộ dữ liệu được trình bày trong Bảng 4.3.

- Bộ CiteULike: Bộ dữ liệu về quản lý tài liệu tham khảo khoa học. Bộ CiteULike chứa các nội dung mô tả dài, trung bình khoảng 66.6 từ trên một mô tả.
- Bộ MovieLens 1M: Bộ dữ liệu đánh giá phim của người dùng. Bộ Movielens chứa các nội dung đánh giá của người dùng là các mô tả ngắn, trung bình chỉ khoảng 4.7 từ trên một mô tả.

[3]Bộ dữ liệu được lấy từ http://www.citeulike.org/faq/data.adp
[4]Bộ dữ liệu được lấy từ https://grouplens.org/datasets/movielens/1m/

Bộ dữ liệu	Số người dùng	Số sản phẩm	Số xếp hạng	Độ dài TB mô tả
CiteULike	5,551	16,890	204,986	66.6
MovieLens 1M	6,040	3,952	1,000,209	4.7

Bảng 4.3: Thống kê các bộ dữ liệu thực nghiệm. Độ thưa thớt biểu thị tỷ lệ của các sản phẩm không có bất kỳ xếp hạng tích cực nào trong mỗi ma trận xếp hạng R.

STT	Tham số cố định	Tham số Bernoulli p	Tham số khảo sát
1	$\lambda = 1000,\ K = 100$	$p = 0.9$	$\alpha \in \{1, 0.1, 0.01, 0.001, 0.0001\}$
2	$\alpha = 0.01,\ \lambda = 1000$	$p = 0.9$	$K \in \{50, 100, 150, 200, 250\}$
3	$\lambda = 1000,\ K = 100$	$p = 0.7$	$\alpha \in \{1, 0.1, 0.01, 0.001, 0.0001\}$
4	$\alpha = 1,\ K = 100$	$p = 0.7$	$\lambda \in \{1, 10, 100, 1000, 10000\}$
5	$\alpha = 1,\ \lambda = 1000$	$p = 0.7$	$K \in \{50, 100, 150, 200, 250\}$

Bảng 4.4: Các kịch bản khảo sát thực nghiệm của chúng tôi. Mô hình CTMP phụ thuộc vào tham số tiên nghiệm Dirichlet α, tham số λ và số chủ đề K.

b. Độ đo đánh giá thực nghiệm

Các dự đoán được đánh giá theo độ đo Precision và Recall cho tất cả người dùng trong bộ thực nghiệm, được đo từ đề xuất top $-M$. Đề xuất top $-M$ chứa các sản phẩm có xếp hạng dự đoán nằm trong số M cao nhất. Để thuận tiện, Precision và Recall tại top-M được viết tắt lần lượt là pre@M và rec@M được định nghĩa:

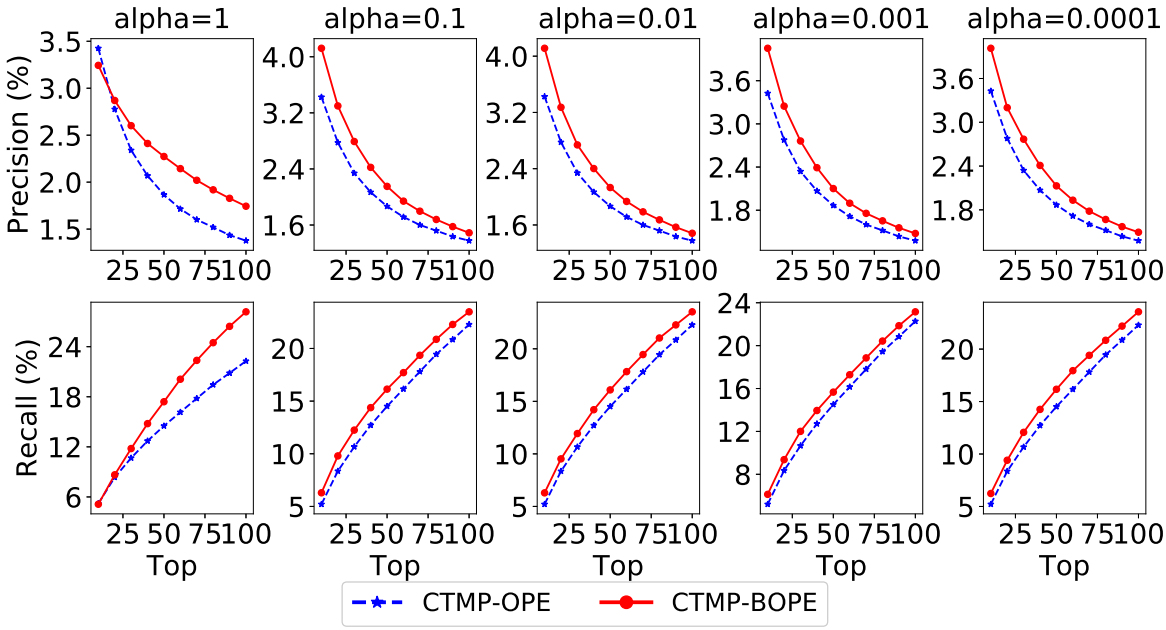
$$prec@M = \frac{1}{U} \sum_{u} \frac{M_u^c}{M}$$
$$rec@M = \frac{1}{U} \sum_{u} \frac{M_u^c}{M_u}$$

trong đó M_u^c là số sản phẩm chính xác xuất hiện trong đề xuất top $-M$ cho người dùng u và M_u là số sản phẩm mà người dùng u đã đánh giá tích cực. Chúng tôi dùng cách đánh giá chéo và ghi lại độ đo Precision và Recall trung bình trên toàn bộ người dùng.

c. Kết quả thực nghiệm

Mặc dù các tác giả trong [115] đã chỉ ra mô hình CTMP tốt hơn mô hình CTF và CTPF, chúng tôi nghiên cứu CTMP và nhận thấy có thể cải tiến CTMP tốt hơn nữa bằng cách áp dụng thuật toán BOPE để suy diễn tham số θ_j trong mô hình CTMP thay vì thuật toán OPE. Đồng thời chúng tôi xem xét tính hiệu quả của BOPE thông qua việc khảo sát ảnh hưởng của tham số tiên nghiệm Dirichlet α, tham số λ và số chủ đề K trong mô hình CTMP.

Chúng tôi cố định tham số $\lambda = 1000$, số chủ đề $K = 100$, khảo sát tham số tiên nghiệm Dirichlet $\alpha \in \{1, 0.1, 0.01, 0.001, 0.0001\}$. Kết quả thực nghiệm được mô tả từ Hình 4.7 đến Hình 4.10.



Hình 4.7: Ảnh hưởng của tham số tiên nghiệm Dirichlet α đến mô hình CTMP khi sử dụng OPE và BOPE suy diễn và tiến hành trên bộ CiteULike. Chúng tôi thiết lập tham số $\lambda = 1000$, số chủ đề $K = 100$ và tham số Bernoulli $p = 0.9$. Độ đo càng cao càng tốt.

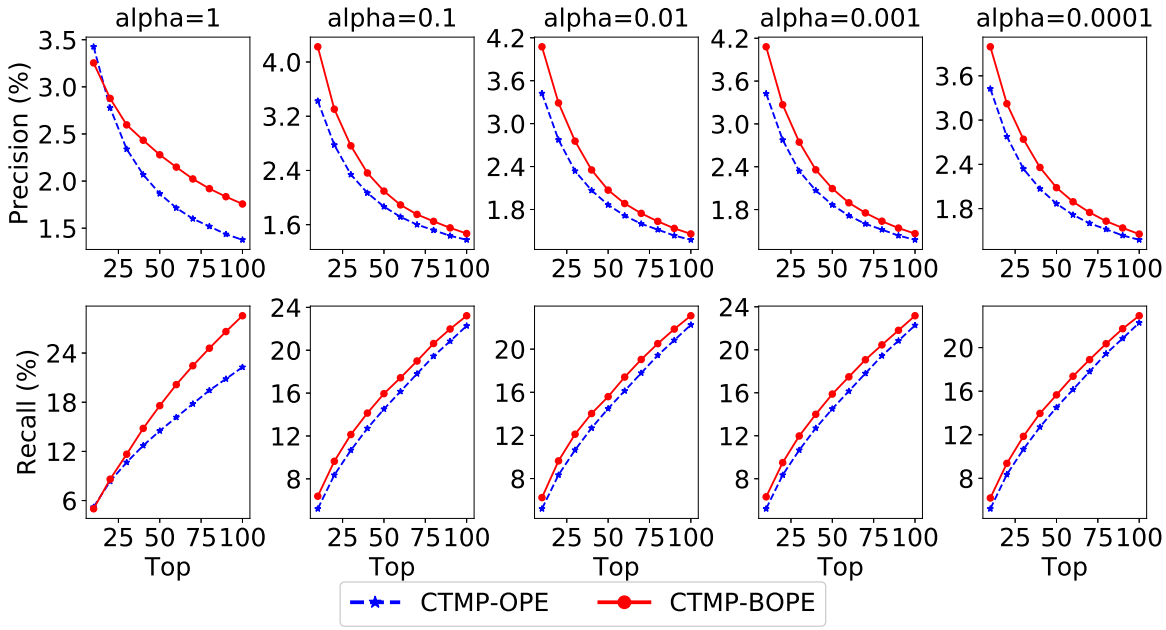
Thông qua các Hình 4.7–4.10 chúng tôi thấy vai trò của tham số α giúp kiểm soát độ thưa của chủ đề hỗn hợp θ cho mỗi nội dung. Nhận thấy CTMP tương đối ổn định khi thay đổi $\alpha \in \{0.1, 0.01, 0.001, 0.0001\}$ trên cả hai tập dữ liệu với hai thuật toán suy diễn OPE và BOPE. Chúng tôi thiết lập mô hình CTMP xác định với tham số tiên nghiệm Dirichlet $\alpha = 1$, tham số $\lambda = 1000$ và số chủ đề $K = 100$, thì khi đó sử dụng thuật toán suy diễn BOPE cho kết quả tốt hơn OPE trên hai độ đo và trên hai tập dữ liệu. Đây là bằng chứng về hiệu quả của thuật toán BOPE khi áp dụng trong các hệ thống gợi ý.

Chúng tôi thiết lập tham số tiên nghiệm Dirichlet $\alpha = 1$, số chủ đề $K = 100$ và chọn tham số Bernoulli $p = 0.7$, sau đó thay đổi tham số $\lambda \in \{1, 10, 100, 1000, 10000\}$. Kết quả thực nghiệm được trình bày từ trong Hình 4.11 và 4.12.

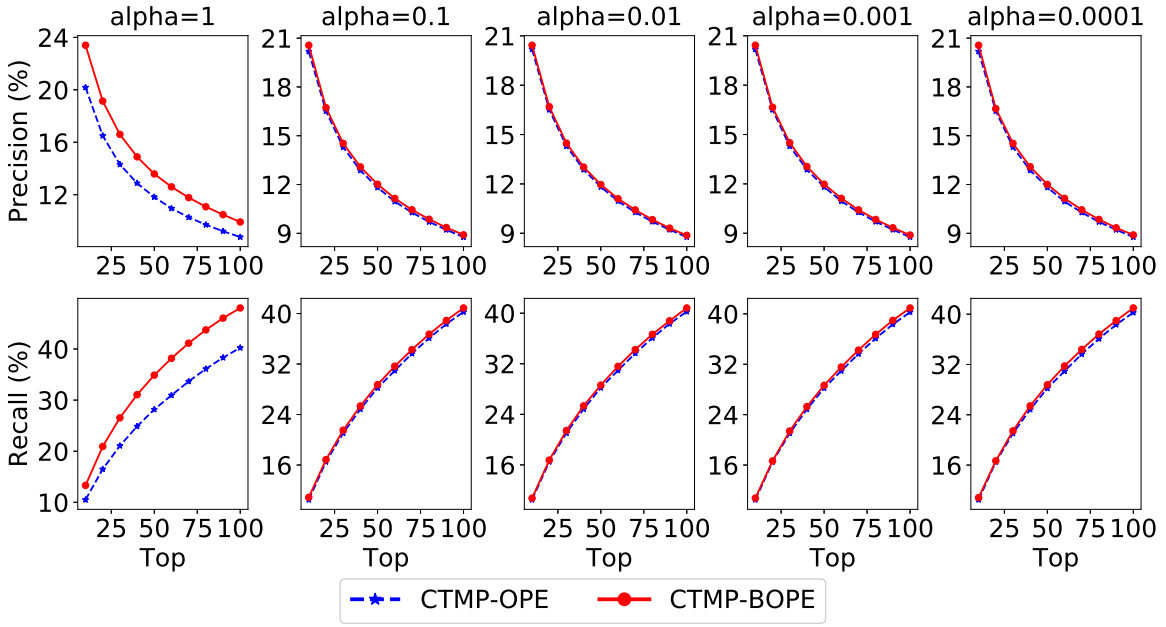
Lưu ý rằng λ là một tham số đặc trưng cho dao động của μ quanh $\boldsymbol{\theta}$. Qua Hình 4.11 và 4.12, chúng tôi thấy rằng khi thiết lập siêu tham số $\alpha = 1$ và $K = 100$, mô hình CTMP tốt hơn với trường hợp $\lambda = 1$ và $\lambda = 10$, trong trường hợp $\lambda = 1000$ hoặc $\lambda = 10000$ thì mô hình cho kết quả xấu đi. Đồng thời chúng tôi thấy rằng với các λ thực nghiệm thì CTMP-BOPE luôn cho kết quả tốt hơn CTMP-OPE, thậm chí trong trường hợp xấu $\lambda = 1000$ hay $\lambda = 10000$.

Để điều tra ảnh hưởng của số chủ đề K đến mô hình CTMP, chúng tôi thiết lập tham số tiên nghiệm Dirichlet $\alpha = 0.01$, tham số $\lambda = 1000$ và chọn tham số Bernoulli $p = 0.9$, sau đó thay đổi số chủ đề $K \in \{50, 100, 150, 200\}$. Những kết quả thực nghiệm này được mô tả trong Hình 4.13 và Hình 4.14.

Chúng tôi tiếp tục tiến hành điều tra sự ảnh hưởng của số chủ đề K khi thiết



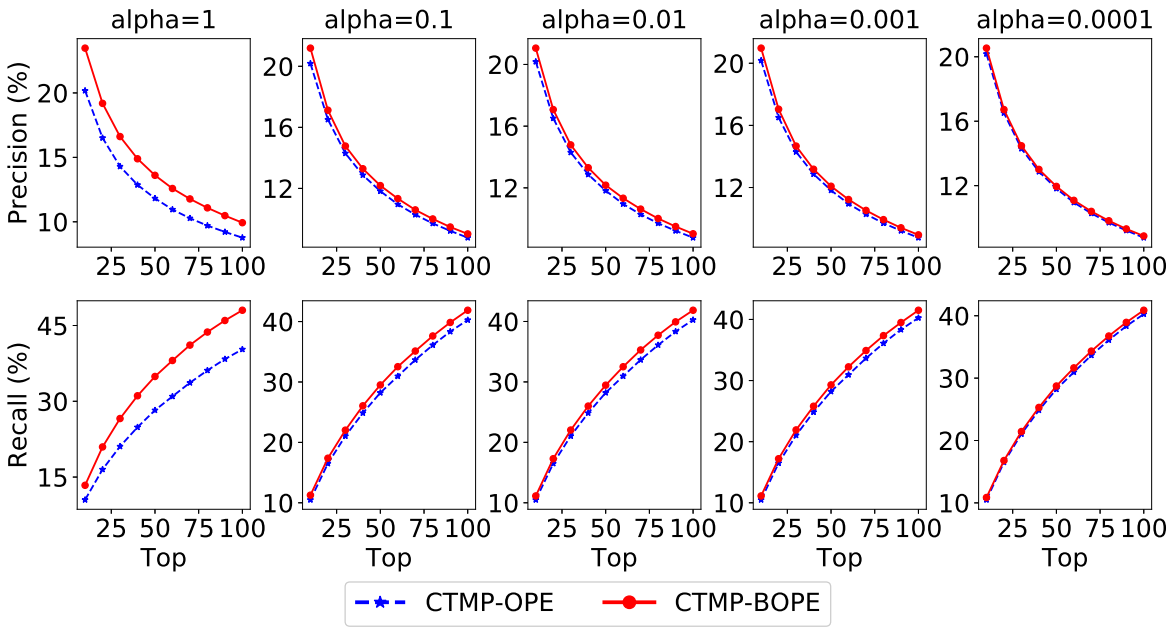
Hình 4.8: Ảnh hưởng của tham số tiên nghiệm Dirichlet α đến mô hình CTMP khi sử dụng OPE và BOPE suy diễn và tiến hành trên bộ CiteULike. Chúng tôi thiết lập tham số $\lambda = 1000$, số chủ đề $K = 100$ và tham số Bernoulli $p = 0.7$ trong BOPE. Độ đo càng cao càng tốt.



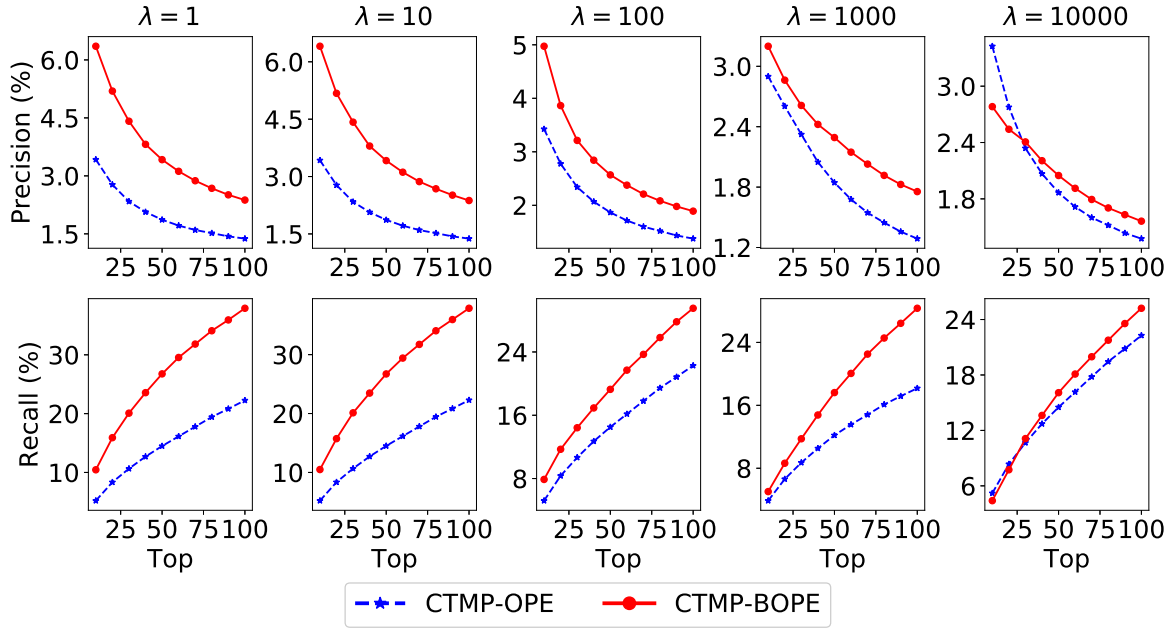
Hình 4.9: Ảnh hưởng của tham số tiên nghiệm Dirichlet α đến mô hình CTMP khi sử dụng OPE và BOPE là thuật toán suy diễn và tiến hành trên bộ dữ liệu MovieLens 1M. Chúng tôi thiết lập tham số $\lambda = 1000$, số chủ đề $K = 100$ và tham số Bernoulli $p = 0.9$. Độ đo càng cao càng tốt.

lập tham số tiên nghiệm Dirichlet $\alpha = 1$, tham số $\lambda = 1000$ và chọn tham số Bernoulli $p = 0.7$. Chúng tôi thay đổi số chủ đề $K \in \{50, 100, 150, 200, 250\}$. Các kết quả thực nghiệm này được mô tả trong Hình 4.15 và 4.16.

Thông qua Hình 4.15 và Hình 4.16 thấy rằng ảnh hưởng của số chủ đề K rõ ràng hơn so với α và λ trong mô hình CTMP. Số lượng chủ đề ẩn K thể hiện sự

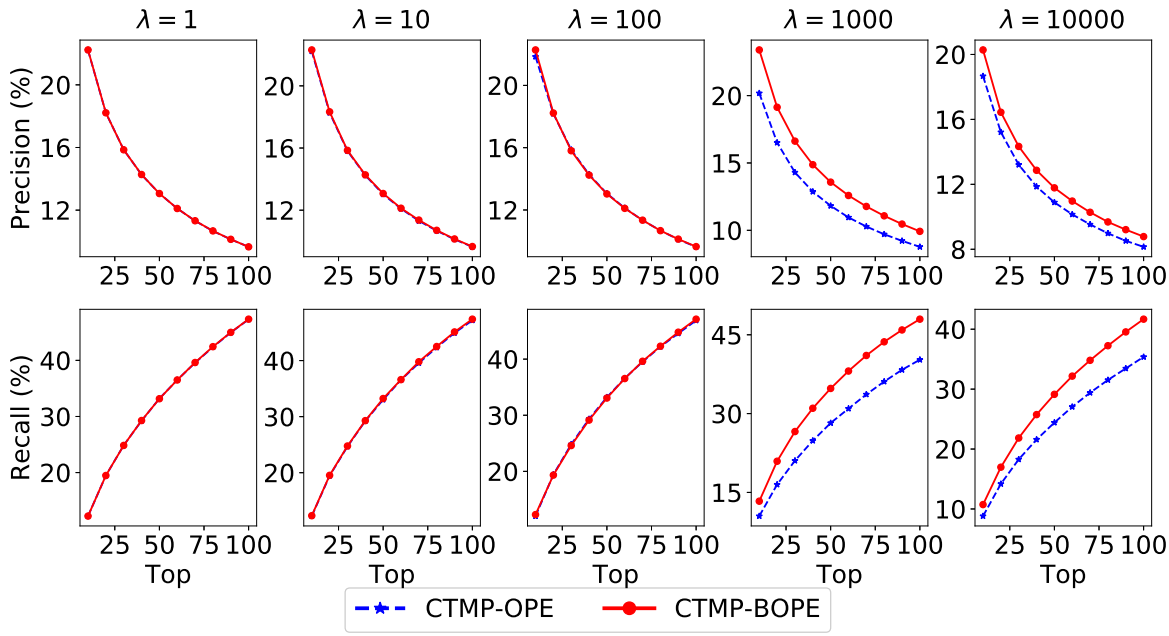


Hình 4.10: Ảnh hưởng của tham số tiên nghiệm Dirichlet α đến mô hình CTMP khi sử dụng OPE và BOPE là thuật toán suy diễn và thực nghiệm trên bộ dữ liệu MovieLens 1M. Chúng tôi thiết lập tham số $\lambda = 1000$, số chủ đề $K = 100$ và tham số Bernoulli $p = 0.7$. Độ đo càng cao càng tốt.

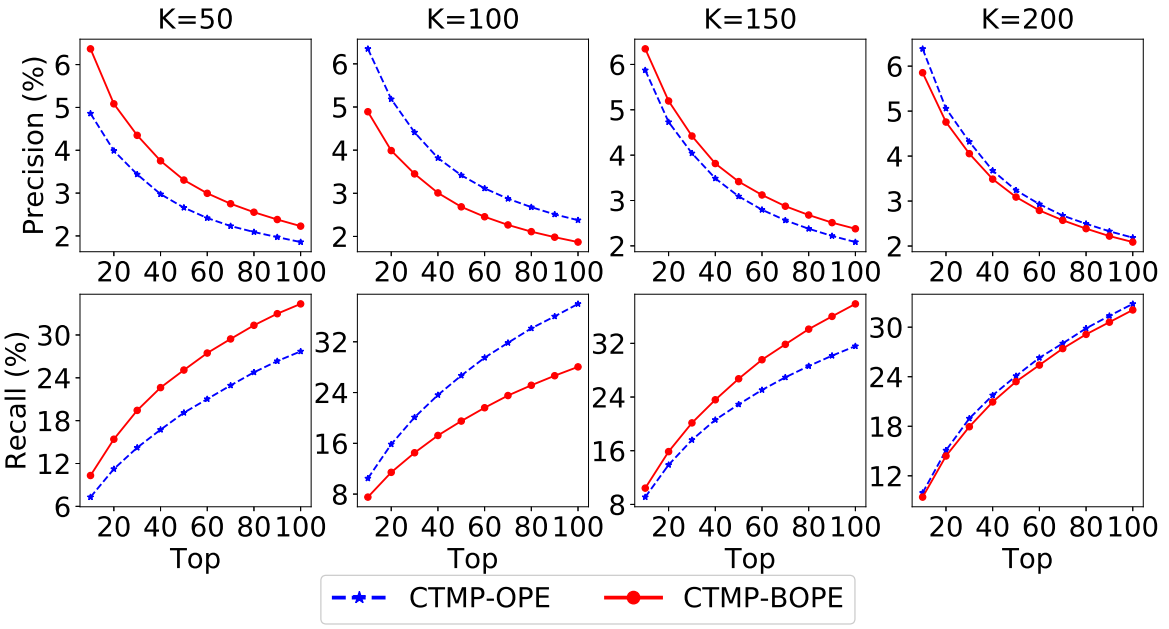


Hình 4.11: Ảnh hưởng của tham số λ đến mô hình CTMP khi sử dụng OPE và BOPE là thuật toán suy diễn và thực nghiệm trên bộ CiteULike. Chúng tôi thiết lập tham số tiên nghiệm Dirichlet $\alpha = 1$, số chủ đề $K = 100$ và tham số Bernoulli $p = 0.7$. Độ đo càng cao càng tốt.

phức tạp của mô hình và phụ thuộc vào tập dữ liệu. Qua các Hình 4.13, 4.14, 4.15 và 4.16, chúng tôi thấy rằng CTMP-BOPE thường tốt hơn CTMP-BOPE. Theo Hình 4.15, CTMP-BOPE đặc biệt tốt hơn CTMP-OPE khi lựa chọn tham số Bernoulli $p = 0.7$ và số chủ đề $K = 200$ hoặc $K = 250$ và trên bộ dữ liệu CiteULike.

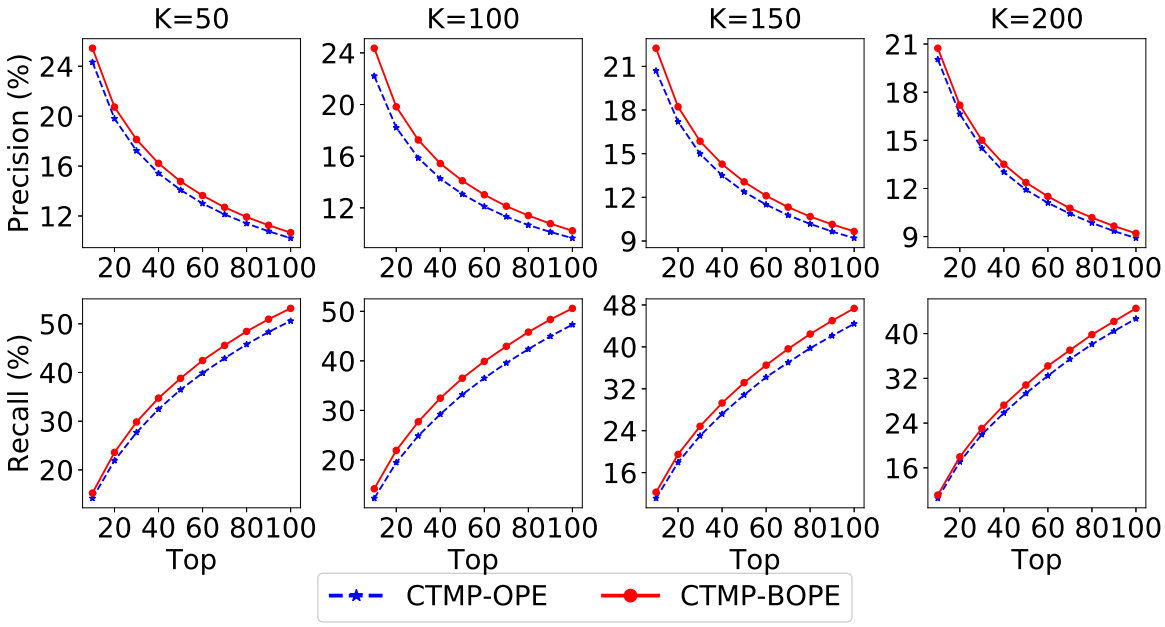


Hình 4.12: Ảnh hưởng của tham số λ đến mô hình CTMP khi sử dụng OPE và BOPE là thuật toán suy diễn và thực nghiệm trên bộ MovieLens 1M. Chúng tôi thiết lập tham số tiên nghiệm Dirichlet $\alpha = 1$, số chủ đề $K = 100$ và tham số Bernoulli $p = 0.7$. Độ đo càng cao càng tốt.

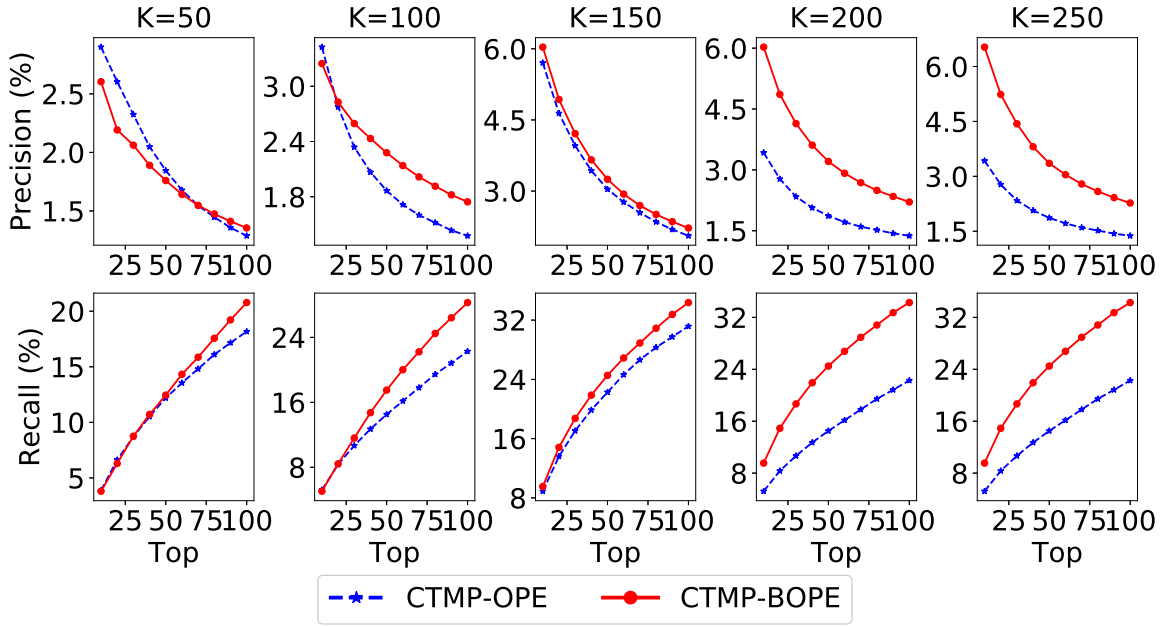


Hình 4.13: Ảnh hưởng của số chủ đề K đến mô hình CTMP khi sử dụng OPE và BOPE làm phương pháp suy diễn và tiến hành trên CiteULike. Chúng tôi thiết lập tham số tiên nghiệm Dirichlet $\alpha = 0.01$, tham số $\lambda = 1000$ và tham số Bernoulli $p = 0.9$. Độ đo càng cao càng tốt.

Mô hình CTMP được đặc trưng bởi các tham số tiên nghiệm Dirichlet α, tham số λ và số chủ đề K. Khi các tham số này thay đổi chúng ta nhận được một mô hình CTMP khác. Điều tra sự thay đổi của mỗi tham số này chúng tôi tiến hành cố định hai tham số còn lại. Kết quả thực hiện mô hình CTMP-OPE và CTMP-BOPE được cho dưới đây. Trong các thực nghiệm này chúng tôi cố



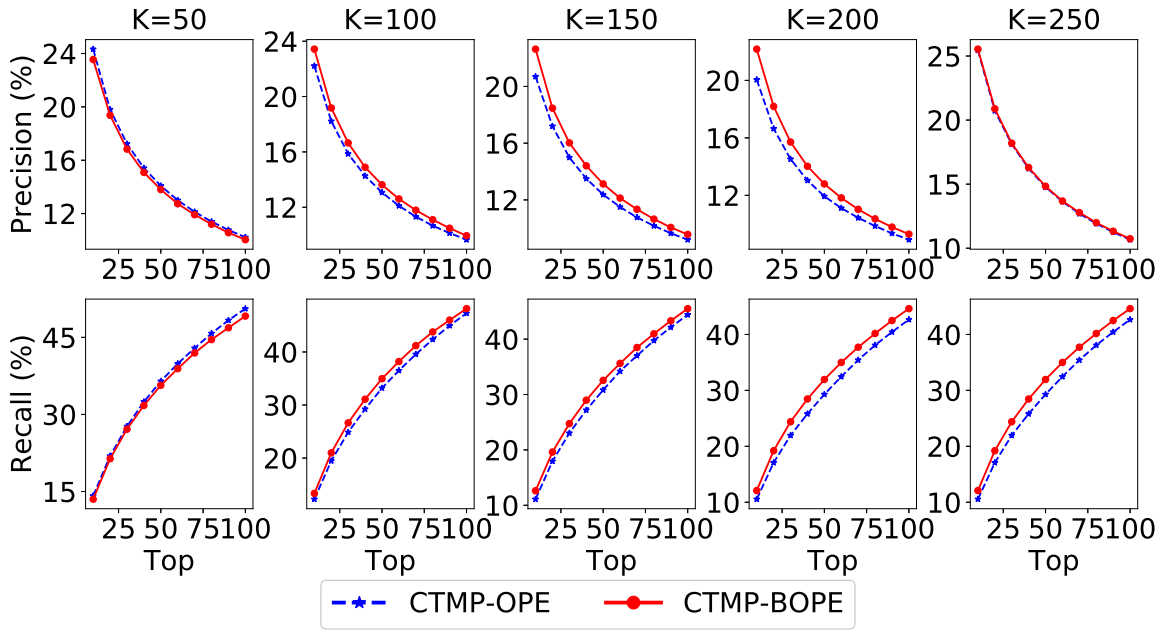
Hình 4.14: Ảnh hưởng của số chủ đề K đến mô hình CTMP khi sử dụng OPE và BOPE làm phương pháp suy diễn và tiến hành trên bộ MovieLens 1M. Chúng tôi thiết lập tham số tiên nghiệm Dirichlet trước $\alpha = 0.01$, tham số $\lambda = 1000$ và tham số Bernoulli $p = 0.9$. Độ đo càng cao càng tốt.



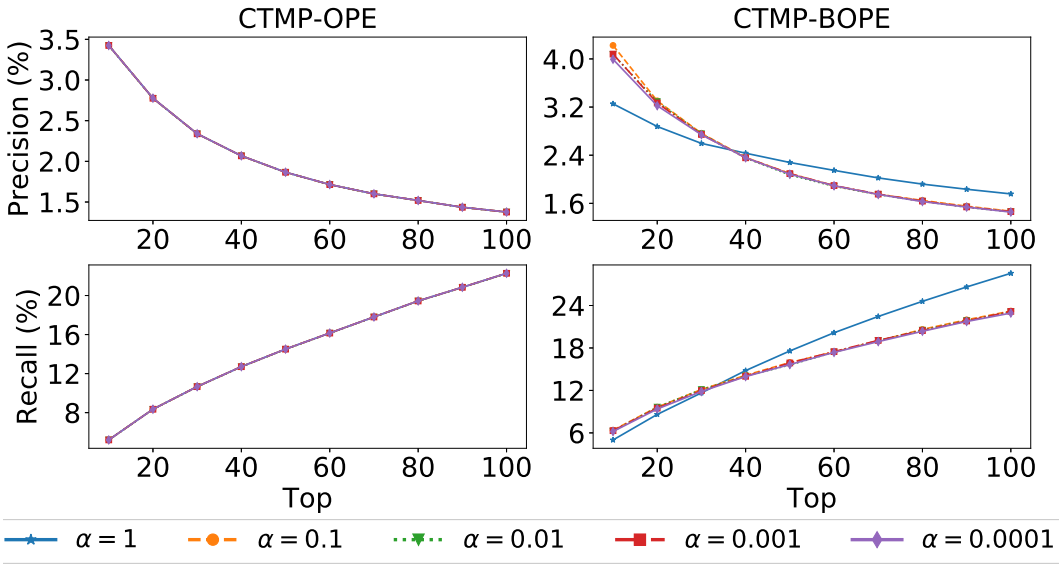
Hình 4.15: Ảnh hưởng của số chủ đề K đến mô hình CTMP khi sử dụng OPE và BOPE là phương pháp suy diễn và tiến hành trên CiteULike. Chúng tôi thiết lập tham số tiên nghiệm Dirichlet $\alpha = 1$, tham số $\lambda = 1000$ và tham số Bernoulli $p = 0.7$. Độ đo càng cao càng tốt.

định tham số Bernoulli là $p = 0.7$ trong thuật toán BOPE.

Thông qua Hình 4.17 và Hình 4.18 chúng tôi thấy sử dụng BOPE làm cho mô hình CTMP có sự khác biệt nhiều hơn khi thay đổi tham số tiên nghiệm Dirichlet α so với OPE ban đầu. Điều đó có được chỉ có thể lý giải là do sự có mặt của tham số phân phối Bernoulli p và chiến lược hai biên ngẫu nhiên trong



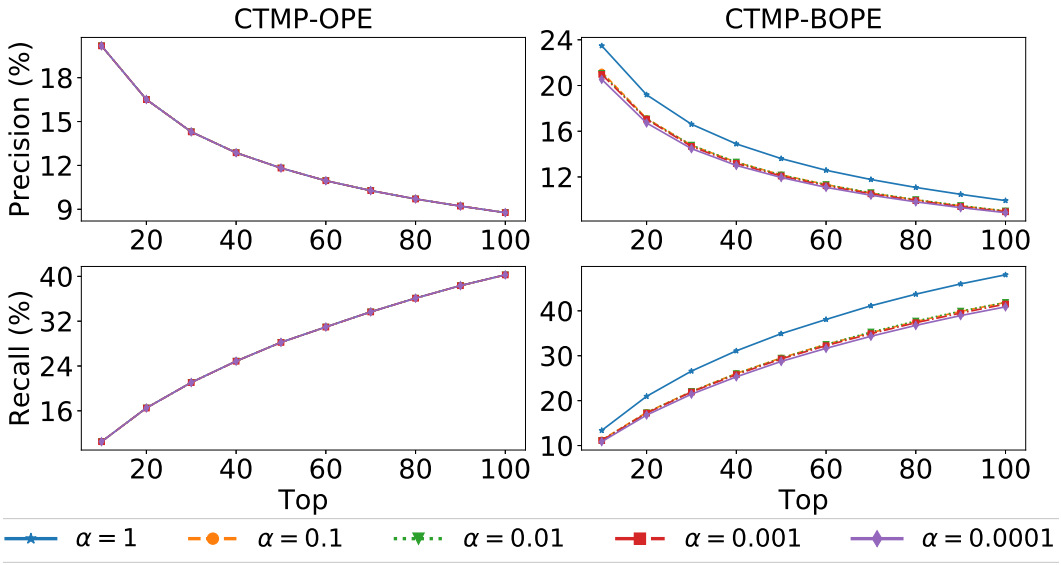
Hình 4.16: Ảnh hưởng của số chủ đề K đến mô hình CTMP khi sử dụng OPE và BOPE là phương pháp suy diễn và tiến hành trên bộ MovieLens 1M. Chúng tôi thiết lập tham số tiên nghiệm Dirichlet $\alpha = 1$, tham số $\lambda = 1000$ và tham số Bernoulli $p = 0.7$. Độ đo càng cao càng tốt.



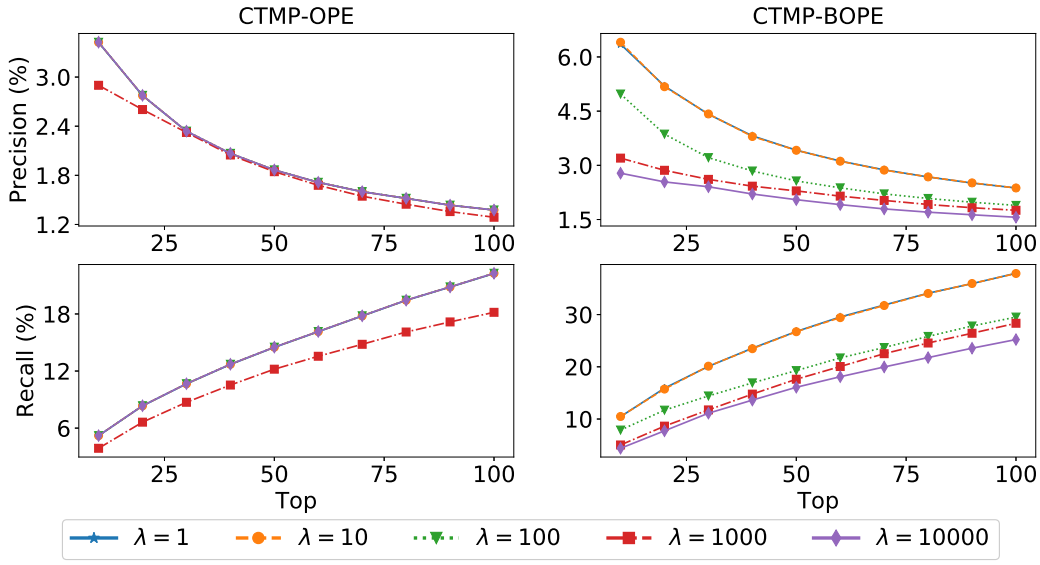
Hình 4.17: Cố định $\lambda = 1000$, số chủ đề $K = 100$ và thay đổi tham số tiên nghiệm Dirichlet $\alpha \in \{1, 0.1, 0, 01, 0.001, 0.0001\}$. Chúng tôi thực nghiệm trên bộ CiteULike và tham số Bernoulli được chọn $p = 0.7$ trong BOPE. Độ đo càng cao càng tốt.

thiết kế BOPE. Theo như các kết quả ở trên, nhận thấy với tham số tiên nghiệm Dirichlet $\alpha = 1$ cho kết quả thường tốt, nên chúng tôi cố định $\alpha = 1$ và số chủ đề $K = 100$ và khảo sát tham số λ.

Theo Hình 4.19 và Hình 4.20 chúng tôi thấy lựa chọn tham số $\lambda = 1000$ hoặc $\lambda = 10000$ không tốt bằng $\lambda = 1$ hoặc $\lambda = 10$. Mô hình có độ đo thấp khi hệ số λ quá lớn, chẳng hạn $\lambda = 10000$. Đồng thời, BOPE với tham số Bernoulli $p = 0.7$



Hình 4.18: Cố định $\lambda = 1000$, số chủ đề $K = 100$ và thay đổi tham số tiên nghiệm Dirichlet $\alpha \in \{1, 0.1, 0, 01, 0.001, 0.0001\}$. Chúng tôi thực nghiệm trên bộ Movielens 1M và tham số Bernoulli được chọn $p = 0.7$ trong BOPE. Độ đo càng cao càng tốt.

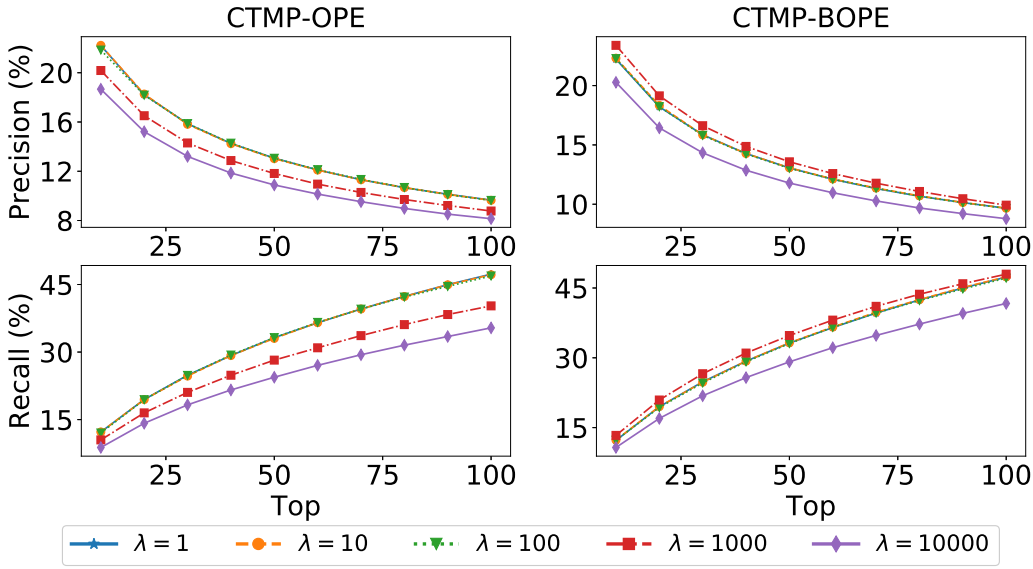


Hình 4.19: Cố định tham số tiên nghiệm Dirichlet $\alpha = 1$, số chủ đề $K = 100$ và thay đổi tham số $\lambda \in \{1, 10, 100, 1000, 10000\}$. Chúng tôi thực nghiệm trên bộ CiteULike và tham số Bernoulli được chọn $p = 0.7$ trong BOPE. Độ đo càng cao càng tốt.

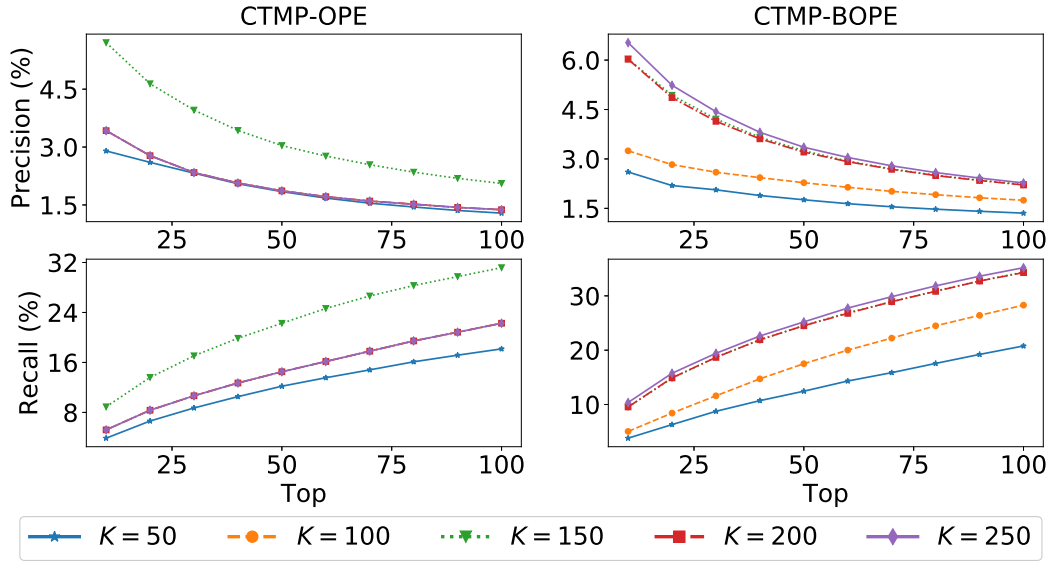
làm cho độ đo của CTMP-BOPE có sự khác biệt lớn khi thay đổi λ và hơn nữa cho kết quả cao hơn hẳn CTMP-OPE, đặc biệt trên bộ CiteULike.

Từ Hình 4.21 và 4.22 chúng tôi thấy độ đo của mô hình tốt hơn khi $K = 250$. Chú ý rằng với mỗi bộ tham số tiên nghiệm Dirichlet $\alpha = 1$, tham số λ và số chủ đề K cho chúng ta một mô hình CTMP cụ thể.

Chúng tôi phát hiện mô hình CTMP sử dụng BOPE (được gọi là CTMP-BOPE) thường hoạt động tốt hơn CTMP sử dụng OPE (được gọi là CTMP-OPE) trên cả hai độ đo Precision và Recall. Đây là bằng chứng trực quan về các



Hình 4.20: Cố định tham số tiên nghiệm Dirichlet $\alpha = 1$, số chủ đề $K = 100$ và thay đổi tham số $\lambda \in \{1, 10, 100, 1000, 10000\}$. Chúng tôi thực nghiệm trên bộ Movielens 1M và tham số Bernoulli được chọn $p = 0.7$ trong BOPE. Độ đo càng cao càng tốt.

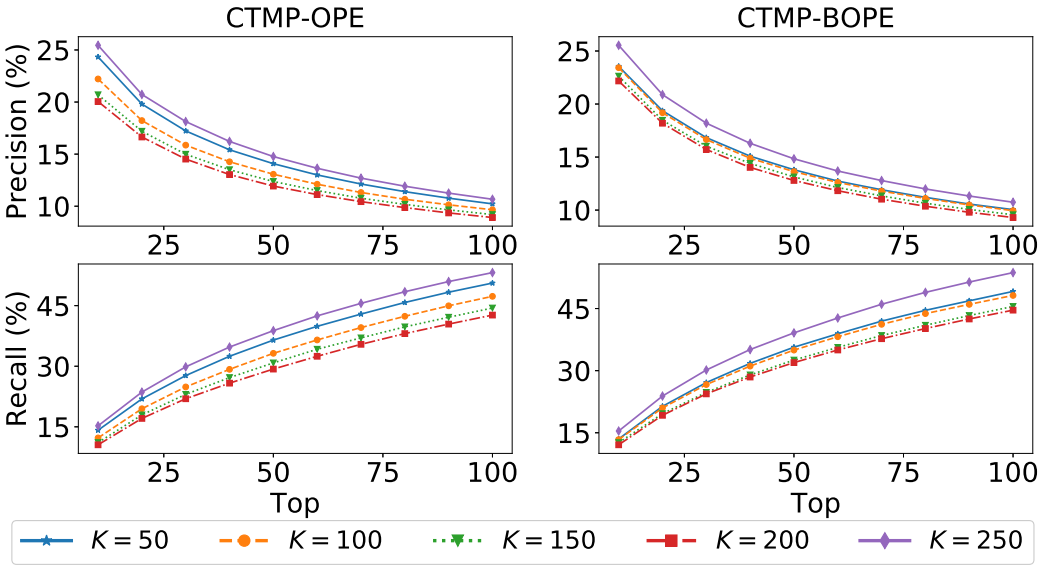


Hình 4.21: Cố định tham số tiên nghiệm Dirichlet $\alpha = 1$, $\lambda = 1000$ và thay đổi số chủ đề $K \in \{50, 100, 150, 200, 250\}$. Chúng tôi thực nghiệm trên bộ CiteULike và tham số Bernoulli được chọn $p = 0.7$ trong BOPE. Độ đo càng cao càng tốt.

ưu điểm của BOPE với việc sử dụng phân phối Bernoulli và hai giới hạn ngẫu nhiên khi so sánh với các phương pháp đã đề xuất trước đây.

4.5. Kết luận chương 4

Trong chương này, chúng tôi đã trình bày về thuật toán BOPE thông qua khai thác tính ngẫu nhiên của phân phối Bernoulli để giải bài toán MAP trong mô hình đồ thị xác suất. BOPE vẫn đảm bảo chất lượng và tốc độ hội tụ giống



Hình 4.22: Cố định tham số tiên nghiệm Dirichlet $\alpha = 1$, $\lambda = 1000$ và thay đổi số chủ đề $K \in \{50, 100, 150, 200, 250\}$. Chúng tôi thực nghiệm trên bộ Movielens 1M và tham số Bernoulli được chọn $p = 0.7$ trong BOPE. Độ đo càng cao càng tốt.

như OPE, đó là đặc điểm quan trọng nhất trong số các phương pháp suy diễn hiện đại. Thông qua các kết quả thực nghiệm, chúng tôi đã chứng minh BOPE có hiệu quả trong bài toán phân tích văn bản và bài toán hệ thống gợi ý. Chúng tôi chứng minh được tham số Bernoulli p trong BOPE có vai trò quan trọng giúp BOPE có những ưu điểm nổi bật như tính hiệu chỉnh và tính linh hoạt tốt, làm việc được trên nhiều loại dữ liệu văn bản, đặc biệt là văn bản ngắn. Hơn nữa BOPE giúp hệ thống giảm hay tránh hiện tượng quá khớp. Với bằng chứng đưa ra về mặt lý thuyết và thực nghiệm, chúng tôi xác nhận rằng BOPE là một ứng cử viên tốt cho bài toán MAP không lồi và hoàn toàn mở rộng cho bài toán tối ưu không lồi tổng quát. Một số kết quả đề cập trong chương này đã được chúng tôi trình bày trong bài báo "A fast algorithm for posterior inference with latent Dirichlet allocation" đăng trên kỷ yếu hội thảo quốc tế ACIIDS 2018 và trong bài báo "Bernoulli randomness in MAP estimation, and its application to text analysis and recommender systems" chuẩn bị gửi đăng trên tạp chí quốc tế uy tín.

KẾT LUẬN

Trong luận án chúng tôi đã nghiên cứu về bài toán cực đại hóa xác suất hậu nghiệm (MAP) không lồi thường xuất hiện trong học máy. Qua đó chúng tôi đã tìm hiểu các cách tiếp cận giải bài toán MAP không lồi. Trên cơ sở đó, luận án đã đề xuất được một số thuật toán ngẫu nhiên giải hiệu quả bài toán MAP không lồi trong một số mô hình xác suất. Sự hiệu quả của các thuật toán đề xuất được xem xét đầy đủ trên cả hai khía cạnh lý thuyết và thực nghiệm. Các thuật toán đề xuất được chứng minh đảm bảo hội tụ với tốc độ nhanh thông qua công cụ ý thuyết xác suất thống kê và lý thuyết tối ưu. Thông qua thực nghiệm triển khai bài toán suy diễn hậu nghiệm trong mô hình chủ đề trên năm bộ dữ liệu lớn và triển khai bài toán MAP với mô hình CTMP trong hệ gợi ý, chúng tôi đảm bảo rằng các đề xuất hiệu quả cao hơn và có khả năng áp dụng tốt so với các phương pháp đương đại. Thông qua nghiên cứu kỹ lưỡng về mặt lý thuyết và thực nghiệm đã chứng minh được tính ưu việt của các thuật toán đề xuất.

A. Kết quả đạt được của luận án

Với kết cấu luận án gồm 4 chương, các kết quả chính đạt được của luận án có thể được tóm tắt như sau:

(1) Luận án đề xuất một nhóm thuật toán tối ưu ngẫu nhiên đặt tên là OPE1, OPE2, OPE3 và OPE4 dựa trên phân phối đều cùng với kết hợp hai biên ngẫu nhiên để giải bài toán suy diễn hậu nghiệm với mô hình chủ đề, trong đó OPE3 và OPE4 là hiệu quả nhất. Sự hội tụ của OPE3 và OPE4 được chứng minh nghiêm túc bằng công cụ giải tích, lý thuyết xác suất và tối ưu.

(2) Chúng tôi tiếp tục đề xuất GOPE bằng sử dụng phân phối rời rạc Bernoulli và lý thuyết xấp xỉ ngẫu nhiên để giải bài toán MAP không lồi. Thuật toán GOPE có tính linh hoạt và tổng quát do có mặt của tham số Bernoulli $p \in (0, 1)$ đóng vai trò là tham số hiệu chỉnh của thuật toán. Chúng tôi đã đánh giá sự hiệu quả của GOPE khi áp dụng cho bài toán MAP với mô hình chủ đề đầy đủ trên hai phương diện lý thuyết và thực nghiệm với dữ liệu đầu vào lớn và cao chiều.

(3) Đề xuất thuật toán BOPE là một thuật toán ngẫu nhiên hiệu quả có tính tổng quát, linh hoạt cao vượt trội hơn các thuật toán khác, đặc biệt là hiệu

chỉnh. Thông qua khai thác ngẫu nhiên Bernoulli và các biến ngẫu nhiên, chúng tôi đã thu được thuật toán BOPE cho bài toán MAP không lồi trong các mô hình đồ thị xác suất. Đồng thời BOPE được áp dụng thành công vào bài toán phân tích văn bản và bài toán hệ gợi ý.

Với các đề xuất chúng tôi thấy rằng các đề xuất đáp ứng tốt các yêu cầu của thuật toán tối ưu cho bài toán không lồi xuất hiện trong học máy: cách vận hành thuật toán đơn giản, thích nghi tốt với nhiều mô hình thực tế, có tốc độ hội tụ nhanh đã được khẳng định thông qua cơ sở lý thuyết và so sánh thực nghiệm.

B. Định hướng phát triển

Các thuật toán tối ưu ngẫu nhiên đề xuất để giải bài toán MAP không lồi được chúng tôi nghiên cứu đem đến một cách tiếp cận mới mẻ: sử dụng xấp xỉ ngẫu nhiên, các phân phối xác suất ngẫu nhiên, đưa hàm mục tiêu tất định ban đầu trở thành đại lượng ngẫu nhiên có thể tính toán hiệu quả. Nhận thấy cách tiếp cận này phù hợp và thực sự hiệu quả, đặc biệt khi các bài toán MAP không lồi trong học máy thống kê thường có hàm mục tiêu phức tạp, xuất hiện trong các mô hình với dữ liệu lớn, cao chiều. Do đó trong thời gian tới, chúng tôi tiếp tục tập trung phát triển các thuật toán sâu và rộng hơn, theo các hướng:

- Triển khai rộng trên nhiều mô hình bài toán khác trong học máy có dạng không lồi hay bài toán quy hoạch DC khó giải;
- Nghiên cứu các tính chất ưu việt của các thuật toán đề xuất như tính tổng quát, tính hiệu quả và khả năng hiệu chỉnh. Từ đó nghiên cứu thuật toán toàn diện hơn trên hai mặt lý thuyết và thực nghiệm;
- Áp dụng thành công vào một số bài toán ứng dụng như phân tích văn bản, hệ gợi ý, bài toán nhận dạng trong xử lý ảnh,... Đồng thời phát triển các nghiên cứu không chỉ làm việc trên các dữ liệu văn bản mà có thể mở rộng trên nhiều loại dữ liệu đa dạng và phức tạp hơn đáp ứng tốt hơn các nhu cầu của các bài toán thực tế.

DANH MỤC CÁC CÔNG TRÌNH ĐÃ CÔNG BỐ CỦA LUẬN ÁN

1. **Xuan Bui**, Tu Vu, and Khoat Than (2016). *Stochastic bounds for inference in topic models*. In International Conference on Advances in Information and Communication Technology (pp. 582-592). Springer, Cham.

2. **Bui Thi-Thanh-Xuan**, Vu Van-Tu, Atsuhiro Takasu, and Khoat Than (2018). *A fast algorithm for posterior inference with latent Dirichlet allocation*. In Asian Conference on Intelligent Information and Database Systems (pp. 137-146). Springer, Cham.

3. Tu Vu, **Xuan Bui**, Khoat Than, and Ryutaro Ichise (2018). *A flexible stochastic method for solving the MAP problem in topic models*, Computación y Sistemas journal, 22(4), 2018 (Scopus, ESCI)

4. **Xuan Bui**, Tu Vu, and Khoat Than (2018). *Some methods for posterior inference in topic models*, Journal Research and Development on Information and Communication Technology (RD-ICT), Vol E-2, No.15 (Tạp chí Công nghệ thông tin và truyền thông)

5. Khoat Than, **Xuan Bui**, Tung Nguyen-Trong, Khang Truong, Son Nguyen, Bach Tran, Linh Ngo, and Anh Nguyen-Duc (2019). *How to make a machine learn continuously: a tutorial of the Bayesian approach*, Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications, 110060I, SPIE.

TÀI LIỆU THAM KHẢO

[1] Pfanzagl J. (2011). *Parametric statistical theory*. Walter de Gruyter.

[2] Dempster A.P., Laird N.M., and Rubin D.B. (1977). *Maximum likelihood from incomplete data via the em algorithm*. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):pp. 1–38.

[3] Seo S., Oh S.D., and Kwak H.Y. (2019). *Wind turbine power curve modeling using maximum likelihood estimation method*. *Renewable energy*, 136:pp. 1164–1169.

[4] Lauritzen S., Uhler C., Zwiernik P., et al. (2019). *Maximum likelihood estimation in gaussian models under total positivity*. *The Annals of Statistics*, 47(4):pp. 1835–1863.

[5] Matilainen K., Mäntysaari E.A., and Strandén I. (2019). *Efficient monte carlo algorithm for restricted maximum likelihood estimation of genetic parameters*. *Journal of Animal Breeding and Genetics*, 136(4):pp. 252–261.

[6] Risk B.B., Matteson D.S., and Ruppert D. (2019). *Linear non-gaussian component analysis via maximum likelihood*. *Journal of the American Statistical Association*, 114(525):pp. 332–343.

[7] Hoffman L.D. and Bradley G.L. (2010). *Calculus for business, economics, and the social and life sciences*. McGraw-Hill.

[8] Boyd S. and Vandenberghe L. (2004). *Convex optimization*. Cambridge University Press.

[9] Bottou L. (1998). *Online learning and stochastic approximations*. *Online learning in Neural Networks*, 17(9):p. 142.

[10] Gauvain J.L. and Lee C.H. (1994). *Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains*. *IEEE transactions on speech and audio processing*, 2(2):pp. 291–298.

[11] Wu M.C.K., Deniz F., Prenger R.J., and Gallant J.L. (2018). *The unified maximum a posteriori (map) framework for neuronal system identification*. *arXiv preprint arXiv:1811.01043*.

[12] Dempster A.P., Laird N.M., and Rubin D.B. (1977). *Maximum likelihood from incomplete data via the em algorithm*. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):pp. 1–22.

[13] Zhang J., Schwing A., and Urtasun R. (2014). *Message passing inference for large scale graphical models with high order potentials*. In *Advances in Neural Information Processing Systems*, pp. 1134–1142.

[14] Darwiche A. (2003). *A differential approach to inference in bayesian networks*. *Journal of the ACM (JACM)*, 50(3):pp. 280–305.

[15] Tosh C. and Dasgupta S. (2019). *The relative complexity of maximum likelihood estimation, map estimation, and sampling*. *Proceedings of Machine Learning Research vol*, 99:pp. 1–43.

[16] Murphy K. (2001). *An introduction to graphical models*. *Rap. tech*, 96:pp. 1–19.

[17] Peyrard N., Cros M.J., de Givry S., Franc A., Robin S., Sabbadin R., Schiex T., and Vignes M. (2019). *Exact or approximate inference in graphical models: why the choice is dictated by the treewidth, and how variable elimination can be exploited*. *Australian & New Zealand Journal of Statistics*, 61(2):pp. 89–133.

[18] Raiffa H. and Schlaifer R. (1972). *Applied statistical decision theory*. In *Applied statistical decision theory*. MIT Press.

[19] Rossi R.J. (2018). *Mathematical Statistics: An Introduction to Likelihood Based Inference*. John Wiley & Sons.

[20] Joshi S. and Miller M.I. (1993). *Maximum a posteriori estimation with good's roughness for three-dimensional optical-sectioning microscopy*. *JOSA A*, 10(5):pp. 1078–1085.

[21] Bassett R. and Deride J. (2019). *Maximum a posteriori estimators as a limit of bayes estimators*. *Mathematical Programming*, 174(1-2):pp. 129–144.

[22] Hazan T., Orabona F., Sarwate A.D., Maji S., and Jaakkola T.S. (2019). *High dimensional inference with random maximum a-posteriori perturbations*. *IEEE Transactions on Information Theory*.

[23] Bereyhi A., Müller R.R., and Schulz-Baldes H. (2019). *Statistical mechanics of map estimation: General replica ansatz*. *IEEE Transactions on Information Theory*.

[24] Siddhu V. (2019). *Maximum a posteriori probability estimates for quantum tomography*. *Physical Review A*, 99(1):p. 012342.

[25] Helin T. and Burger M. (2015). *Maximum a posteriori probability estimates in infinite-dimensional bayesian inverse problems*. *Inverse Problems*, 31(8):p. 085009.

[26] Kodamana Z.L.H. and Huang A.A.B. (2019). *A gmm-mrf based image segmentation approach for interface level estimation*. *IFAC-PapersOnLine*, 52(1):pp. 28–33.

[27] Pereyra M. (2019). *Revisiting maximum-a-posteriori estimation in log-concave models*. *SIAM Journal on Imaging Sciences*, 12(1):pp. 650–670.

[28] Than K. and Doan T. (2015). *Guaranteed algorithms for inference in topic models*. *arXiv preprint arXiv:1512.03308*.

[29] Than K., Bui X., Nguyen-Trong T., Truong K., Nguyen S., Tran B., Ngo L., and Nguyen-Duc A. (2019). *Can machines learn continuously? a tutorial of the bayesian approach*. In *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*. SPIE.

[30] Jameel S., Fu Z., Shi B., Lam W., and Schockaert S. (2019). *Word embedding as maximum a posteriori estimation*. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 6562–6569.

[31] D'Ambrosio B. (1999). *Inference in bayesian networks*. *AI magazine*, 20(2):pp. 21–21.

[32] Hoffman M.D., Blei D.M., Wang C., and Paisley J.W. (2013). *Stochastic variational inference.*. *Journal of Machine Learning Research*, 14(1):pp. 1303–1347.

[33] Blei D.M., Kucukelbir A., and McAuliffe J.D. (2016). *Variational inference: A review for statisticians*. *Journal of the American Statistical Association, to appear*.

[34] Neal R.M. (1993). *Probabilistic inference using Markov chain Monte Carlo methods*. Department of Computer Science, University of Toronto Toronto, Ontario, Canada.

[35] Chib S. (2003). *Monte carlo methods and bayesian computation: Overview. se fienberg, jb kadane, eds*. *International Encyclopedia of the Social and Behavioral Sciences: Statistics*.

[36] Bottou L., Curtis F.E., and Nocedal J. (2018). *Optimization methods for large-scale machine learning*. *Siam Review*, 60(2):pp. 223–311.

[37] Sontag D. and Roy D. (2011). *Complexity of inference in latent dirichlet allocation*. In *Proceedings of Advances in Neural Information Processing System*.

[38] Gill J. and Heuberger S. (2019). *Bayesian modeling and inference: A postmodern perspective*. *LC Curini & J. Franzese, Robert J., eds,'Handbook of Research Methods in Political Science & International Relations', Sage*.

[39] Blei D.M., Ng A.Y., and Jordan M.I. (2003). *Latent dirichlet allocation*. *Journal of machine Learning research*, 3:pp. 993–1022.

[40] Teh Y.W., Newman D., and Welling M. (2006). *A collapsed variational bayesian inference algorithm for latent dirichlet allocation*. In *Proceedings of Advances in Neural Information Processing Systems*, pp. 1353–1360.

[41] Teh Y.W., Kurihara K., and Welling M. (2007). *Collapsed variational inference for hdp*. In *Proceedings of Advances in Neural Information Processing Systems*, pp. 1481–1488.

[42] Asuncion A., Welling M., Smyth P., and Teh Y.W. (2009). *On smoothing and inference for topic models*. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pp. 27–34. AUAI Press.

[43] Hoffman M., Blei D.M., and Mimno D.M. (2012). *Sparse stochastic inference for latent dirichlet allocation*. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pp. 1599–1606. ACM.

[44] Yuille A.L. and Rangarajan A. (2003). *The concave-convex procedure*. *Neural computation*, 15(4):pp. 915–936.

[45] Mairal J. (2013). *Stochastic majorization-minimization algorithms for large-scale optimization*. In *Advances in Neural Information Processing Systems*, pp. 2283–2291.

[46] Clarkson K.L. (2010). *Coresets, sparse greedy approximation, and the frank-wolfe algorithm*. *ACM Trans. Algorithms*, 6(4):pp. 1–30.

[47] Hazan E. and Kale S. (2012). *Projection-free online learning*. In *Proceedings of Annual International Conference on Machine Learning*.

[48] Swoboda P. and Kolmogorov V. (2019). *Map inference via block-coordinate frank-wolfe algorithm*. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11146–11155.

[49] Dai B., He N., Dai H., and Song L. (2016). *Provable bayesian inference via particle mirror descent*. In *Artificial Intelligence and Statistics*, pp. 985–994.

[50] Simsekli U., Badeau R., Cemgil T., and Richard G. (2016). *Stochastic quasi-newton langevin monte carlo*. In *International Conference on Machine Learning*.

[51] Than K. and Ho T.B. (2012). *Fully sparse topic models*. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 490–505. Springer.

[52] Than K. and Ho T.B. (2015). *Inference in topic models: sparsity and trade-off*. *arXiv preprint arXiv:1512.03300*.

[53] Anandkumar A. and Ge R. (2015). *Efficient approaches for escaping higher order saddle points in non-convex optimization*. In *Conference on Learning Theory*, pp. 797–842.

[54] Gelman A., Carlin J.B., Stern H.S., Dunson D.B., Vehtari A., and Rubin D.B. (2013). *Bayesian data analysis*. Chapman and Hall/CRC.

[55] Tuy H. (2016). *Motivation and overview*. In *Convex Analysis and Global Optimization*, pp. 127–149. Springer.

[56] Robbins H. and Monro S. (1951). *A stochastic approximation method*. *The Annals of Mathematical Statistics*, pp. 400–407.

[57] Xiao L. and Zhang T. (2014). *A proximal stochastic gradient method with progressive variance reduction*. *SIAM Journal on Optimization*, 24(4):pp. 2057–2075.

[58] Blake A. and Zisserman A. (1987). *Visual reconstruction*. MIT press.

[59] Hazan E., Levy K.Y., and Shalev-Shwartz S. (2016). *On graduated optimization for stochastic non-convex problems*. In *International Conference on Machine Learning*, pp. 1833–1841.

[60] Chen X., Liu S., Sun R., and Hong M. (2018). *On the convergence of a class of adam-type algorithms for non-convex optimization*. *arXiv preprint arXiv:1808.02941*.

[61] Duchi J., Hazan E., and Singer Y. (2011). *Adaptive subgradient methods for online learning and stochastic optimization*. *Journal of Machine Learning Research*, 12:pp. 2121–2159.

[62] Tieleman T. and Hinton G. (2012). *Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude*. *COURSERA: Neural networks for Machine learning*, 4(2):pp. 26–31.

[63] Zeiler M.D. (2012). *Adadelta: an adaptive learning rate method*. *arXiv preprint arXiv:1212.5701*.

[64] Kingma D.P. and Ba J.L. (2014). *Adam: Amethod for stochastic optimization*. In *Proc. 3rd Int. Conf. Learn. Representations*.

[65] Ghadimi S. and Lan G. (2016). *Accelerated gradient methods for nonconvex nonlinear and stochastic programming*. *Mathematical Programming*, 156(1-2):pp. 59–99.

[66] Allen-Zhu Z. (2018). *Natasha 2: Faster non-convex optimization than sgd*. In *Advances in Neural Information Processing Systems*, pp. 2680–2691. Curran Associates, Inc.

[67] Allen-Zhu Z. and Li Y. (2018). *Neon2: Finding local minima via first-order oracles*. In *Advances in Neural Information Processing Systems*, pp. 3720–3730.

[68] Pascanu R., Dauphin Y.N., Ganguli S., and Bengio Y. (2014). *On the saddle point problem for non-convex optimization*. *arXiv preprint arXiv:1405.4604*.

[69] Dauphin Y.N., Pascanu R., Gulcehre C., Cho K., Ganguli S., and Bengio Y. (2014). *Identifying and attacking the saddle point problem in high-dimensional non-convex optimization*. In *Advances in Neural Information Processing Systems*, pp. 2933–2941.

[70] Ge R., Huang F., Jin C., and Yuan Y. (2015). *Escaping from saddle points—online stochastic gradient for tensor decomposition*. In *Conference on Learning Theory*, pp. 797–842.

[71] Jin C., Ge R., Netrapalli P., Kakade S.M., and Jordan M.I. (2017). *How to escape saddle points efficiently*. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1724–1732. JMLR. org.

[72] Reddi S.J., Sra S., Póczos B., and Smola A. (2016). *Stochastic frank-wolfe methods for nonconvex optimization*. In *54th Annual Allerton Conference on Communication, Control, and Computing*, pp. 1244–1251. IEEE.

[73] Lei L., Ju C., Chen J., and Jordan M.I. (2017). *Non-convex finite-sum optimization via scsg methods*. In *Advances in Neural Information Processing Systems*, pp. 2348–2358.

[74] Jordan M.I. and Bishop C. (2004). *An introduction to graphical models*.

[75] Koller D. and Friedman N. (2009). *Probabilistic graphical models: principles and techniques*. MIT press.

[76] Zhang N.L. and Poole D. (1994). *A simple approach to bayesian network computations*. In *Proceedings of the Biennial Conference-Canadian Society for Computational Studies of Intelligence*, pp. 171–178.

[77] Cozman F.G. et al. (2000). *Generalizing variable elimination in bayesian networks*. In *Workshop on Probabilistic reasoning in Artificial intelligence*, pp. 27–32. Editora Tec Art São Paulo, Brazil.

[78] Chavira M. and Darwiche A. (2007). *Compiling bayesian networks using variable elimination.*. In *IJCAI*, pp. 2443–2449.

[79] Attias H. (2000). *A variational bayesian framework for graphical models*. In *Advances in Neural Information Processing Systems*, pp. 209–215.

[80] Bishop C.M. (2006). *Pattern recognition and Machine learning*. springer.

[81] Blei D.M., Kucukelbir A., and McAuliffe J.D. (2017). *Variational inference: A review for statisticians*. *Journal of the American Statistical Association*, 112(518):pp. 859–877.

[82] Minka T. and Lafferty J. (2002). *Expectation-propagation for the generative aspect model*. In *Proceedings of the Eighteenth conference on Uncertainty in Artificial intelligence*, pp. 352–359. Morgan Kaufmann Publishers Inc.

[83] Carlo M.C.M. (2006). *stochastic simulation for bayesian inference*. *CRC Texts in Statistical Science Series*.

[84] Parisi G. (1988). *Statistical field theory*. Addison-Wesley.

[85] Geman S. and Geman D. (1987). *Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images*. Elsevier.

[86] Hastings W.K. (1970). *Monte carlo sampling methods using markov chains and their applications*. *Biometrika*, 57(1):pp. 97–109.

[87] DeGroot M.H. (2005). *Optimal statistical decisions*, volume 82. John Wiley & Sons.

[88] Green P.J., Łatuszyński K., Pereyra M., and Robert C.P. (2015). *Bayesian computation: a summary of the current state, and samples backwards and forwards*. *Statistics and Computing*, 25(4):pp. 835–862.

[89] Bottou L. and Vapnik V. (1992). *Local learning algorithms*. *Neural Computation*, 4(6):pp. 888–900.

[90] Scott Deerwester S.T., George W T.K., and Harshman R. (1990). *Indexing by latent semantic analysis*. *Journal of The American society for information science*, 41(6).

[91] Hoffman T. (1999). *Probabilistic latent semantic indexing*. *Annual international conference on Research and development in information retrieval*.

[92] Griffiths T.L. and Steyvers M. (2004). *Finding scientific topics*. In *Proceedings of the National academy of Sciences*, volume 101, pp. 5228–5235. National Acad Sciences.

[93] Mimno D., Hoffman M., and Blei D. (2012). *Sparse stochastic inference for latent dirichlet allocation*. In *29th Annual International Conference on Machine Learning*.

[94] Frank M. and Wolfe P. (1956). *An algorithm for quadratic programming*. *Naval Research Logistics*, 3(1-2):pp. 95–110.

[95] Land A.H. and Doig A.G. (1960). *An automatic method of solving discrete programming problems*. *Econometrica: Journal of the Econometric Society*, pp. 497–520.

[96] Le Thi H.A. and Pham Dinh T. (2005). *The dc (difference of convex functions) programming and dca revisited with dc models of real world nonconvex optimization problems*. *Annals of Operations Research*, 133(1-4):pp. 23–46.

[97] Than K. and Doan T. (2015). *Dual online inference for latent dirichlet allocation*. In *Asian Conference on Machine Learning*, pp. 80–95.

[98] Hoffman M., Bach F.R., and Blei D.M. (2010). *Online learning for latent dirichlet allocation*. In *advances in Neural Information Processing Systems*, pp. 856–864.

[99] Bottou L. and Bousquet O. (2007). *Learning using large datasets.*. In *NATO ASI Mining Massive Data Sets for Security*, pp. 15–26. Citeseer.

[100] Foulds J., Boyles L., DuBois C., Smyth P., and Welling M. (2013). *Stochastic collapsed variational bayesian inference for latent dirichlet allocation*. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data mining*, pp. 446–454. ACM.

[101] Bottou L. (1999). *On-line learning and stochastic approximations*. In *On-line learning in neural networks*, pp. 9–42. Cambridge University Press.

[102] Aletras N. and Stevenson M. (2013). *Evaluating topic coherence using distributional semantics*. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013)*, pp. 13–22. Association for Computational Linguistics.

[103] Feller W. (1943). *The general form of the so-called law of the iterated logarithm*. *Transactions of the American Mathematical Society*, 54(3):pp. 373–402.

[104] An L.T.H. (2003). *Dc programming for solving a class of global optimization problems via reformulation by exact penalty*. In *Global Optimization and Constraint Satisfaction: First International Workshop on Global Constraint Optimization and Constraint Satisfaction, COCOS 2002, Valbonne-Sophia Antipolis, France, October 2002. Revised Selected Papers 1*, pp. 87–101. Springer.

[105] De Moivre A. (2001). *The doctrine of chances*. In *Annotated Readings in the History of Statistics*, pp. 32–36. Springer.

[106] Robert C. (2007). *The Bayesian choice: from decision-theoretic foundations to computational implementation*. Springer Science & Business Media.

[107] Reddi S.J., Sra S., Póczos B., and J.Smola A. (2016). *Stochastic frank-wolfe methods for nonconvex optimization*. In *Proceedings of 54th Annual Allerton Conference on Communication, Control, and Computing*, pp. 1244–1251. IEEE.

[108] Box G.E., Hunter J.S., and Hunter W.G. (2005). *Statistics for experimenters*. In *Wiley Series in Probability and Statistics*. Wiley Hoboken, NJ, USA.

[109] Sato I. and Nakagawa H. (2015). *Stochastic divergence minimization for online collapsed variational bayes zero inference of latent dirichlet allocation*. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1035–1044. ACM.

[110] Mai K., Mai S., Nguyen A., Van Linh N., and Than K. (2016). *Enabling hierarchical dirichlet processes to work better for short texts at large scale*. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 431–442. Springer.

[111] Tang J., Zhang M., and Mei Q. (2013). *One theme in all views: modeling consensus topics in multiple contexts*. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 5–13. ACM.

[112] Arora S., Ge R., Koehler F., Ma T., and Moitra A. (2016). *Provable algorithms for inference in topic models*. In *International Conference on Machine Learning*, pp. 2859–2867.

[113] Cuong H.N., Tran V.D., Van L.N., and Than K. (2019). *Eliminating overfitting of probabilistic topic models on short and noisy text: The role of dropout*. *International Journal of Approximate Reasoning*.

[114] Dieng A.B., Ruiz F.J., and Blei D.M. (2019). *Topic modeling in embedding spaces*. *arXiv preprint arXiv:1907.04907*.

[115] Le H.M., Cong S.T., The Q.P., Van Linh N., and Than K. (2018). *Collaborative topic model for poisson distributed ratings*. *International Journal of Approximate Reasoning*, 95:pp. 62–76.

[116] Wang C. and Blei D.M. (2011). *Collaborative topic modeling for recommending scientific articles*. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 448–456. ACM.

[117] Gopalan P.K., Charlin L., and Blei D. (2014). *Content-based recommendations with poisson factorization*. In *Advances in Neural Information Processing Systems*, pp. 3176–3184.

[118] Lau J.H., Newman D., and Baldwin T. (2014). *Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality*. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2014, April 26-30, 2014, Gothenburg, Sweden*, pp. 530–539.

Phụ lục

A. Độ đo Log Predictive Probability

Độ đo *Log Predictive Probability* (LPP) cho thấy tính dự đoán và khái quát của mô hình $\mathcal{M}$ trên dữ liệu mới. Việc tính toán phép đo này được thực hiện theo bài báo [43]. Đối với mỗi tài liệu trong bộ dữ liệu thực nghiệm, chia ngẫu nhiên thành hai phần riêng $\mathbf{w}_{obs}$ và $\mathbf{w}_{ho}$ với tỷ lệ $80:20$. Tiếp theo, suy luận về $\mathbf{w}_{obs}$ để có được ước tính $E(\boldsymbol{\theta}^{obs})$. Sau đó, xấp xỉ xác suất dự đoán là

$$P(\mathbf{w}_{ho}|\mathbf{w}_{obs},\mathcal{M}) \simeq \prod_{(w\in\mathbf{w}_{ho})}\sum_{k=1}^{K} E(\boldsymbol{\theta}_k^{obs})E(\boldsymbol{\beta}_{kw}) \tag{A1}$$

$$\textit{Log Predictive Probability} = \log\frac{P(\mathbf{w}_{ho}|\mathbf{w}_{obs},\mathcal{M})}{|\mathbf{w}_{ho}|} \tag{A2}$$

trong đó $\mathcal{M}$ là mô hình cần đo. Ước tính $E(\boldsymbol{\beta}_k) \propto \boldsymbol{\lambda}_k$ cho các phương pháp học tập duy trì phân phối biến phân ($\boldsymbol{\lambda}$) theo các chủ đề. LPP được tính trung bình từ 5 lần chạy ngẫu nhiên, mỗi lần thực hiện kiểm tra trên 1000 tài liệu văn bản.

B. Độ đo Normalised Pointwise Mutual Information

Độ đo Normalised Pointwise Mutual Information (NPMI) giúp chúng ta thấy được sự gắn kết hoặc chất lượng ngữ nghĩa của các chủ đề riêng lẻ. Theo [118], NPMI tốt với đánh giá về tính có thể hiểu của các mô hình chủ đề. Với mỗi chủ đề t, lấy tập $\{w_1, w_2, \ldots, w_n\}$ của top n thuật ngữ với xác suất cao nhất. Sau đó tính:

$$NPMI(t) = \frac{2}{n(n-1)}\sum_{j=2}^{n}\sum_{i=1}^{j-1}\frac{\log\frac{P(w_j,w_i)}{P(w_j)P(w_i)}}{-\log P(w_j,w_i)} \tag{B1}$$

trong đó $P(w_i, w_j)$ là xác suất để term w_i và w_j xuất hiện cùng nhau trong một văn bản. Ước lượng những xác suất này từ tập huấn luyện. Trong các thực nghiệm, chọn top $n = 10$ từ ngữ cho mỗi chủ đề. Toàn bộ NPMI của mô hình với K chủ đề được tính trung bình như sau:

$$NPMI = \frac{1}{K}\sum_{t=1}^{K} NPMI(t) \tag{B2}$$