

TỐI ƯU SỐ LƯỢNG THÔNG BÁO TRONG HỆ THỐNG KHUYẾN NGHỊ TIN TỨC NHỜ TỐI ƯU RÀNG BUỘC

Nguyễn Thị Trang**

*Khoa Công nghệ thông tin, Học viện Công nghệ Bưu chính Viễn thông,

+Công ty cổ phần VCCorp

Tóm tắt: Chức năng thông báo là một chức năng quan trọng của các hệ thống khuyến nghị. Chức năng này giúp tăng cam kết hoặc duy trì hoạt động của người dùng đối với hệ thống. Tuy nhiên, trong trường hợp gửi quá nhiều thông báo có thể dẫn đến gây phiền cho người dùng, thậm chí người dùng xóa tài khoản trên hệ thống. Trong bài báo này, chúng tôi tập trung vào bài toán tối ưu số lượng thông báo gửi đến người dùng trong hệ thống khuyến nghị tin tức. Mỗi người dùng của hệ thống sẽ được cung cấp một ngân sách thông báo trong một thời gian nhất định. Áp dụng tối ưu ràng buộc giúp cân bằng số lượng tin gửi đi dựa trên ràng buộc về số lượng ấn vào bài viết (Click) của tập người dùng. Nghiên cứu này áp dụng thực tế cho 2000 người dùng của hệ thống Lotus. Kết quả chỉ ra số lượng thông báo đã giảm đáng kể trong khi vẫn đảm bảo được chỉ số CTR - *Click Through Rate* (Tỉ lệ Click vào bài viết) của hệ thống cũng như tốt hơn với tập dùng còn lại của hệ thống.

Từ khóa: Hệ thống khuyến nghị gửi thông báo, Tối ưu ràng buộc đa mục tiêu.

I. MỞ ĐẦU

Mục tiêu của các hệ thống khuyến nghị nội dung trực tuyến là xác định được bài báo xu hướng trong hàng ngàn nội dung được đăng tải lên hằng ngày. Nhiều hệ thống hiện tại làm việc dựa trên những phản hồi của người dùng, ví dụ "CTR: Tỉ lệ Click vào bài viết", mục đích là tối đa tỉ lệ này bằng việc dự đoán những nội dung có khả năng người dùng tương tác cao dựa trên dự đoán chỉ số CTR của các bài biết sau đó lựa chọn những bài thích hợp đưa tới người dùng.[1],[2],[3], [4]. Từ những năm 2011, trong nghiên cứu [5], D Agarwal và các đồng tác giả đã đề xuất sử dụng tối ưu ràng buộc cho việc tối ưu hoá thời gian trải nghiệm của người dùng dựa trên thông tin lịch sử Click cho trang tin tức Yahoo. Nghiên cứu chỉ ra rằng vẫn có những đánh đổi đáng kể giữa các mục tiêu. Tuy nhiên, nhóm tác giả chỉ tiếp cận tối ưu hoá theo từng nhóm người dùng (user) vì vậy vẫn chưa giải quyết vấn đề cá nhân hoá tới từng người dùng. Trong nghiên cứu [6], nhóm tác giả đã làm việc với Hàm đối ngẫu Lagrange để có thể cá nhân hoá tới từng người dùng. Tiếp theo, D Agarwal và các đồng tác giả [7] cũng áp dụng tối ưu ràng buộc cho trang "HomePage" của hệ thống Linkedin bằng cách tối ưu các cam kết của người dùng như (Click, Share,...) kết hợp các ràng buộc về kinh tế.

Ngoài những tương tác tích cực, người dùng cũng có những tương tác tiêu cực vì vậy nhóm nghiên cứu của Linkedin [8] cũng áp dụng kĩ thuật này để tối ưu hoá lượng thông báo email tới người dùng dựa trên thông tin ràng buộc từ những tương tác tích cực và tiêu cực.

Lotus là một trang mạng xã hội mới của Việt Nam với cách tiếp cận nội dung. Hiện tại hằng ngày có khoảng hơn 20 nghìn người dùng hoạt động. Hệ thống thông báo của Lotus hoạt động dựa trên việc gửi những tin tức mới nhất và liên quan đến người dùng hằng ngày. Việc gửi thông báo cho người dùng giúp tăng trải nghiệm, khuyến khích người dùng vào hệ thống. Tuy nhiên nếu gửi thông báo quá nhiều, có thể dẫn tới gây phiền cho người dùng, thậm chí khiến người dùng tắt thông báo hoặc xóa app. Như vậy để giải quyết vấn đề này, nghiên cứu này đề xuất một giải pháp giúp ước lượng được số lượng ngân sách tin để gửi thông báo tới người dùng một cách hợp lý. Ý tưởng là có thể dùng các ràng buộc để tối ưu được số lượng tin gửi đến người dùng đến một ngưỡng nào đó sao cho vẫn đảm bảo được tỉ lệ CTR (*click-through-rate*) như mong muốn.

Vì vậy trong nghiên cứu này, tôi tập trung vào tìm hiểu về tối ưu ràng buộc và áp dụng nó vào giải quyết tìm ra ngân sách tin tức cho từng người dùng Lotus. Phần II giới thiệu các định nghĩa các hàm mục tiêu cần tối ưu cho bài toán tối ưu số lượng thông báo. Phần III trình bày những vấn đề cơ bản trong tối ưu có ràng buộc, hơn nữa tôi có trình bày về bài toán Quy hoạch tuyến tính (Linear Programming-LP), một bài toán được áp dụng trong nhiều vấn đề thực tế. Tiếp theo là áp dụng LP để giải quyết vấn đề Tối ưu số lượng thông báo cho hệ thống Lotus. Phần IV trình bày quá trình cài đặt và thử nghiệm thực tế trên một tập người dùng của Lotus. Cuối cùng là phân tích kết quả và đưa ra vấn đề tồn tại cũng như hướng phát triển tương lai.

II. ĐỊNH NGHĨA VẤN ĐỀ

Trong phần này, nghiên cứu giới thiệu về các mục tiêu cần tối ưu trong vấn đề Tối ưu đa mục tiêu (*multi-objective programming* - MOPs). Tại nghiên cứu này xem xét hai mục tiêu : (a) Tổng số Click của người dùng khi nhận được thông báo và (b) Tổng số tin đã gửi cho người dùng trên hệ thống tin tức. Những mục tiêu này còn được coi là những thước đo tương tác chuẩn cho nhiều hệ thống tin tức, chúng đóng một vai trò quan trọng trong sự định hình chiến lược quảng cáo và đo lường hiệu suất cho website. Hơn nữa về khía cạnh sự hài lòng của người dùng, nó cũng đóng vai trò quan trọng, là bước tiền đề để có thêm những tương tác khác vào nội dung sau khi Click như là Thích (Like), Chia

Tác giả liên hệ: Nguyễn Thị Trang,

Email: trangnt@ptit.edu.vn

Đến tòa soạn: 10/2022, chỉnh sửa: 12/2022, chấp nhận đăng: 12/2022.

sẽ (Share) hoặc Bình luận (Comment). Trong một hệ thống khuyến nghị tin tức, chức năng thông báo là một yếu tố quan trọng để duy trì hoạt động của người dùng, tuy nhiên nó cũng thể gây phiền nhiễu dẫn tới người dùng ít vào hệ thống hoặc thậm chí không sử dụng khi số lượng tin tức gửi quá nhiều. Phương thức tiếp cận theo MOPs giúp tránh vấn đề không cân bằng số lượng tin gửi tới người dùng. Trong nghiên cứu này, tác giả sẽ làm việc với cả hai thước đo trên để giải thích cho ý tưởng chính trên hệ thống tin tức thực tế.

Giả sử ta có các định nghĩa sau, N_t thể hiện số lượng người dùng được gửi đi thông báo trong khoảng thời t . Giả sử p_{it} thể hiện tỉ lệ CTR vào thông báo của user i trong thời gian t . x_i thể hiện số lượng ngân sách tin tức bắt ở epoch tiếp theo cho user i . Trong nghiên cứu này, chỉ giải quyết cho từng epoch riêng biệt, vì vậy ở các công thức tôi sẽ không cho t vào.

- **Tổng số tin gửi:** Với mỗi user có một ngân sách số tin tôi đã được gửi là x_i . Như vậy tổng số tin ước lượng được gửi cho N user là:

$$TotalSends(x) = \sum_{i=1}^N x_i \quad (1)$$

- **Tổng số tin Click:** Tổng số tin được Click của N user sau khi nhận được thông báo trong epoch hiện tại.

$$TotalClicks(x) = \sum_{i=1}^N x_i * p_i \quad (2)$$

III. MÔ HÌNH TỐI ƯU

Trong phần này, tôi sẽ trình bày những vấn đề cơ bản trong tối ưu đa mục tiêu hay tối ưu mục tiêu có ràng buộc, chi tiết có thể tham khảo tại [9] và [10].

A. Các khái niệm cơ bản

Bài toán tối ưu dạng tổng quát:

$$\begin{aligned} x^* &= \operatorname{argmin}_x f_0(x) \\ \text{s.t. } f_i(x) &\leq 0, \quad i = 1, 2, \dots, m \\ h_j(x) &= 0, \quad j = 1, 2, \dots, p \end{aligned} \quad (3)$$

Phát biểu bằng lời: Tìm giá trị của biến x để tối thiểu hàm $f_0(x)$ trong số các giá trị của x thỏa mãn điều kiện ràng buộc. Ta có bảng các khái niệm và kí hiệu trong bài toán tối ưu như trong Bảng I. Ngoài ra, khi:

- Khi $m = p = 0$ bài toán trên được gọi là *bài toán tối ưu không ràng buộc*
- $\mathcal{D}$ là tập xác định, tức giao của tất cả tập xác định của mọi hàm số xuất hiện trong bài toán. tập hợp các điểm thỏa mãn mọi điều kiện ràng buộc, thông thường là một tập con của $\mathcal{D}$ được gọi là tập khả thi – (feasible set). Khi tập khả thi là một tập rỗng thì ta nói bài toán tối ưu trên là *vô nghiệm*. Nếu một điểm nằm trong tập khả thi, ta gọi điểm đó là *điểm khả thi*.
- **Giá trị tối ưu – Optimal value** của bài toán tối ưu trên được định nghĩa là:

$$p^* = \inf\{f_0(x) \mid f_i(x) \leq 0, i = 1, \dots, m; h_j(x) = 0, j = 1, \dots, p\}$$

Trong đó $\inf$ là viết tắt của hàm infimum. p^* có thể nhận các giá trị $\pm\infty$. Nếu bài toán là *vô nghiệm*, tức là

không có điểm nào thỏa mãn tất cả ràng buộc, ta coi $p^* = +\infty$. Nếu hàm mục tiêu *không bị chặn dưới* (unbounded below) trong tập xác định, ta coi $p^* = -\infty$

Bảng I: Bảng chú thích kí hiệu

Kí hiệu	Ý nghĩa
$x \in R^n$	Biến tối ưu
$f_0: R^n \rightarrow R$	Hàm mục tiêu
$f_i(x) \leq 0$	Bất đẳng thức ràng buộc
$f_i: R^n \rightarrow R$	Hàm bất đẳng thức ràng buộc
$h_j(x) = 0$	Đẳng thức ràng buộc
$h_j: R^n \rightarrow R$	Hàm đẳng thức ràng buộc
$\mathcal{D} = \bigcap_{i=1}^m \operatorname{dom} f_i \cap \bigcap_{j=1}^p \operatorname{dom} h_j$	Tập xác định

B. Điểm tối ưu và các điểm tối ưu cục bộ

Một điểm $x^* = p^*$ thuộc tập xác định $\mathcal{D}$ được gọi là một *điểm tối ưu*, hoặc là nghiệm của bài toán nếu x^* là điểm khả thi và $x^* = p^*$. Tập hợp hợp tất cả các điểm tối ưu được gọi là *tập điểm tối ưu*. Nếu tập điểm tối ưu là một tập không rỗng, ta nói bài toán trên là *giải được*. Ngược lại, nếu tập điểm tối ưu là tập rỗng, ta nói giá trị tối ưu là *không thể đạt được*.

Ví dụ: Xét hàm mục tiêu $f(x) = \frac{1}{x}$ với ràng buộc $x > 0$. Giá trị tối ưu của bài toán này là $p^* = 0$ nhưng tập điểm tối ưu là một tập rỗng vì không có giá trị nào của x để hàm mục tiêu đạt giá trị 0. Lúc này ta nói giá trị tối ưu là *không đạt được*.

Với hàm một biến, một điểm là *cực tiểu* của hàm số nếu tại đó, hàm số đạt giá trị nhỏ nhất trong một lân cận (và lân cận này thuộc tập xác định của hàm số). Trong không gian một chiều, lân cận của một điểm được hiểu là tập các điểm cách đó một khoảng rất nhỏ. Trong không gian nhiều chiều, ta gọi một điểm x là *điểm tối ưu cục bộ* nếu tồn tại một giá trị $R > 0$ sao cho:

$$\begin{aligned} f_0(x) &= \inf\{f_0(z) \mid f_i(z) \leq 0, i = 1, \dots, m, \\ h_j(z) &= 0, j = 1, \dots, p; \|z - x\|_2 \leq R\} \end{aligned} \quad (4)$$

Nếu một điểm khả thi x thỏa mãn $f_i(x) = 0$ ta nói rằng bất đẳng thức ràng buộc thứ i : $f_i(x) = 0$ là *thỏa được*. Nếu i : $f_i(x) < 0$, ta nói ràng buộc này là *không thỏa được* tại x .

Mặc dù trong định nghĩa bài toán tối ưu (3) là cho bài toán *tối thiểu hàm mục tiêu* với các ràng buộc thỏa mãn các điều kiện nhỏ hơn hoặc bằng 0, các bài toán tối ưu với *tối đa hàm mục tiêu* và điều kiện ràng buộc ở dạng khác đều có thể đưa về được dạng này:

$$\max f_0(x) \Leftrightarrow \min -f_0(x) \quad (5)$$

$$f_i(x) \leq g(x) \Leftrightarrow f_i(x) - g(x) \leq 0 \quad (6)$$

$$f_i(x) \geq 0 \Leftrightarrow -f_i(x) \leq 0 \quad (7)$$

$$a \leq f_i(x) \leq b \Leftrightarrow f_i(x) - b \leq 0 \text{ \& } a - f_i(x) \leq 0 \quad (8)$$

$$f_i(x) \leq 0 \Leftrightarrow f_i(x) + s_i = 0, s_i \geq 0 \quad (9)$$

C. Bài toán tối ưu lồi

Trong bài toán tối ưu, chúng ta đặc biệt quan tâm tới những bài toán mà hàm mục tiêu là hàm lồi, và tập khả thi cũng là một tập lồi. Một bài toán tối ưu lồi - (convex optimization problem) là một bài toán tối ưu dạng:

$$\begin{aligned} x^* &= \operatorname{argmin}_x f_0(x) \\ \text{s.t: } f_i(x) &\leq 0; \quad i = 1, 2, \dots, m \\ h_j(x) &= a_j^T x - b_j = 0; \quad j = 1, \dots, p \end{aligned} \quad (10)$$

Trong đó $f_0, f_1, \dots, f_m$ là các hàm lồi. So với bài toán tối ưu, bài toán tối ưu lồi có thêm ba điều kiện nữa:

- Hàm mục tiêu là một hàm lồi.
- Các hàm bất đẳng thức ràng buộc f_i là các hàm lồi.
- Hàm đẳng thức ràng buộc h_i là affine.

Ta có vài nhận xét như sau:

- Tập hợp các điểm thỏa mãn $h_j(x) = 0$ là một tập lồi vì nó có dạng một siêu phẳng.
- Khi f_i là một hàm lồi thì tập hợp các điểm thỏa mãn $f_i(x) \leq 0$ chính là đường đồng mức 0 của f_i và là một tập lồi.

Như vậy, tập hợp các điểm thỏa mãn mọi điều kiện ràng buộc chính là giao điểm của các tập lồi, vì vậy nó là một tập lồi. Từ đó ta có thể kết luận rằng, trong bài toán tối ưu lồi, ta tối thiểu một hàm mục tiêu lồi trên một tập lồi.

Tính chất quan trọng nhất của bài toán tối ưu lồi chính là mọi điểm tối ưu cục bộ chính là một điểm tối ưu toàn cục (điểm cực tiểu chính là nghiệm của bài toán). Việc này chúng mình bằng phản chứng. Gọi x_0 là một điểm cực bộ:

$$f_0(x) = \inf\{f_0(x) \mid x \in \text{Tập khả thi}, \|x - x_0\|_2 \leq R\} \quad (11)$$

Với $R > 0$ nào đó. Giả sử x_0 không phải là một điểm tối ưu toàn cục, tức tồn tại một điểm khả thi y sao cho $f(y) < f(x_0)$ (hiển nhiên rằng y không nằm trong lân cận đang xét). Ta có thể tìm được $\theta \in [0, 1]$ đủ nhỏ sao cho $z = (1 - \theta)x_0 + \theta y$ nằm trong lân cận của x_0 tức $\|z - x_0\|_2 < R$. Việc này có được vì tập khả thi là một tập lồi. Hơn nữa, vì hàm mục tiêu f_0 là một hàm lồi, ta có:

$$\begin{aligned} f_0(z) &= f_0((1 - \theta)x_0 + \theta y) \\ &\leq (1 - \theta)f_0(x_0) + \theta f_0(y) \\ &< (1 - \theta)f_0(x_0) + \theta f_0(x_0) \\ &= f_0(x_0) \end{aligned} \quad (12)$$

Điều này mâu thuẫn với giả thiết x_0 là một điểm cực tiểu. Vậy giả sử sai, tức x_0 chính là điểm tối ưu toàn cục, ta có điều phải chứng minh.

Điều kiện tối ưu cho hàm mục tiêu khả vi: Nếu hàm mục tiêu f_0 là khả vi, theo first-order condition, với mọi $x, y \in \operatorname{dom} f_0$, ta có:

$$f_0(x) \geq f_0(x_0) + \nabla f_0(x_0)^T (x - x_0) \quad (13)$$

Đặt χ là tập khả thi. Điều kiện cần và đủ để một điểm $x_0 \in \chi$ là điểm tối ưu là:

$$\nabla f_0(x_0)^T (x - x_0) \geq 0, \forall x \in \chi \quad (14)$$

Điều này chỉ ra rằng nếu $\nabla f_0(x_0) = 0$ thì x_0 chính là một điểm tối ưu của bài toán. Nếu $\nabla f_0(x_0) \neq 0$, nghiệm của bài toán sẽ phải nằm trên biên của tập khả thi. Thật vậy, quan sát Hình 1, điều kiện này nói rằng nếu x_0 là một điểm tối ưu thì với $\forall x \in \chi$, vector đi từ hàm mục tiêu tại x_0 tới x hợp với vector $-\nabla f_0(x_0)$ một góc tù. Nói cách khác, nếu ta vẽ mặt tiếp tuyến của hàm mục tiêu tại x_0 thì mọi điểm khả thi nằm về một phía so với mặt tiếp tuyến này. Điều này chỉ ra rằng x_0 phải nằm trên biên của tập khả thi χ . Hơn nữa, tập khả thi nằm về phía làm cho hàm mục tiêu đạt giá trị cao hơn $f_0(x_0)$. Mặt tiếp tuyến này chính là siêu phẳng hỗ trợ của tập khả thi tại điểm x_0 . Nhắc lại rằng khi vẽ các đường đồng mức, chúng ta dùng màu lam để chỉ giá trị nhỏ, màu đỏ để chỉ giá trị lớn của hàm. Trong đó siêu phẳng hỗ trợ là một mặt phẳng đi qua một điểm trên biên của một tập hợp sao cho mọi điểm trong tập hợp đó nằm về một phía (hoặc nằm trên) so với mặt phẳng đó. Nếu một tập hợp lồi, tồn tại một siêu phẳng hỗ trợ tại mọi điểm trên biên của nó.

Trong bài toán tối ưu lồi có ba bài toán cơ bản: Quy hoạch tuyến tính (Linear programming-LP), Quy hoạch toàn phương (Quadratic programming) và Quy hoạch hình học (geometric programming). Trong nghiên cứu này, tác giả sẽ đi sâu vào bài toán LP để áp dụng cho bài toán tối ưu số lượng thông báo cho hệ thống đề xuất tin tức.

D. Dạng quy hoạch tuyến tính-LP

Dạng tổng quát: Ta có dạng tổng quát của Linear programming như sau:

$$\begin{aligned} x^* &= \operatorname{argmin}_x c^T x + d \\ \text{s.t: } Gx &\leq h \\ Ax &= b \end{aligned} \quad (15)$$

Trong đó $G \in R^{m \times n}$, $h \in R^m$, $A \in R^{p \times n}$, $c, x \in R^n$ và $d \in R$. Số vô hướng d chỉ làm thay đổi giá trị của hàm mục tiêu mà không làm thay đổi nghiệm của bài toán nên có thể được lược bỏ. Nhắc lại rằng kí hiệu $\leq$ nghĩa là mỗi phần tử trong vector ở vế trái nhỏ hơn hoặc bằng phần tử tương ứng trong vector ở vế phải. Chú ý rằng nhiều bất đẳng thức dạng $g_i x \leq h_i$, với g_i là các vector hàng, có thể viết gộp dưới dạng $Gx \leq h$ trong đó mỗi hàng của G ứng với một g_i , mỗi phần tử của h tương ứng với một h_i .

Dạng tiêu chuẩn: Trong dạng tiêu chuẩn (standard form) LP, các bất đẳng thức ràng buộc chỉ là điều kiện các nghiệm có thành phần không âm. Dạng tiêu chuẩn sẽ có dạng như công thức dưới đây:

$$\begin{aligned} x^* &= \operatorname{argmin}_x c^T x \\ \text{s.t: } Ax &= b \\ x &\geq 0 \end{aligned} \quad (16)$$

Dạng tổng quát ở Công thức 15 có thể được đưa về dạng tiêu chuẩn (16) bằng cách đặt thêm biến slack s .

$$\begin{aligned} x^* &= \operatorname{argmin}_{x,s} c^T x \\ \text{s.t: } Ax &= b \\ Gx + s &= h \\ s &\geq 0 \end{aligned} \quad (17)$$

Tiếp theo nếu ta biểu diễn x dưới dạng hiệu của hai vector mà thành phần của nó đều không âm, tức $x = x^+ - x^-$, với $x^+, x^- \geq 0$. Ta có thể viết lại công thức 17 như sau:

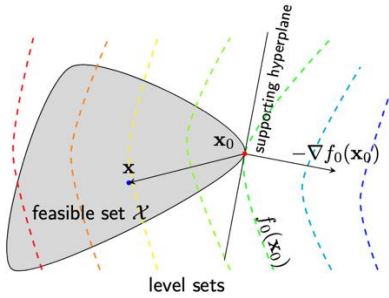
$$\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x}^+, \mathbf{x}^-, s} \mathbf{c}^T \mathbf{x}^+ - \mathbf{c}^T \mathbf{x}^- \quad (18)$$

$$\text{s.t. } \mathbf{A}\mathbf{x}^+ - \mathbf{A}\mathbf{x}^- = \mathbf{b}$$

$$\mathbf{G}\mathbf{x}^+ - \mathbf{G}\mathbf{x}^- + \mathbf{s} = \mathbf{h}$$

$$\mathbf{x}^+ \geq \mathbf{0}, \mathbf{x}^- \geq \mathbf{0}, \mathbf{s} \geq \mathbf{0}$$

Các bài toán LP có thể được minh họa như Hình 1. Điểm $\mathbf{x}_0$ chính là điểm làm cho hàm mục tiêu đạt giá trị nhỏ nhất, điểm $\mathbf{x}_1$ chính là điểm làm cho hàm mục tiêu đạt giá trị lớn nhất. Nghiệm của bài toán LP, nếu có thường là một điểm ở đỉnh của polyhedron feasible set hoặc là một mặt của polyhedron đó (trong trường hợp các đường đồng mức của hàm mục tiêu song song với mặt đó, hàm mục tiêu đạt giá trị tối ưu)



Hình 1: Biểu diễn hình học của điều kiện tối ưu cho hàm mục tiêu khả vi. Các đường nét đứt có màu tương ứng với các level sets (đường đồng mức)

E. Tối ưu số lượng thông báo sử dụng LP

Lấy cảm hứng nghiên cứu từ nghiên cứu [5] đã tối ưu trải nghiệm người dùng dựa vào hai yếu tố là "Tổng số lượt Click" và "Tổng số thời gian trải nghiệm". Ở nghiên cứu này, sử dụng LP giải quyết vấn đề "g-MOP: Global Constrained Optimization-Tối ưu ràng buộc toàn cục" theo công thức sau:

$$\text{maximize } \text{Downstream}(\mathbf{x}) \quad (19)$$

$$\text{s.t. } \text{TotalClicks}(\mathbf{x}) \geq \alpha \text{TotalClicks}^*$$

Trong đó $\text{Downstream}(\mathbf{x})$ ở đây chính là yếu tố "Thời gian trải nghiệm" của người dùng và $\alpha \in [0,1]$. Từ ý tưởng này, tôi có thể áp dụng vào bài toán tối ưu tin tức thông báo cho người dùng. Mục tiêu có thể phát biểu như sau: "Tối thiểu hoá lượng tin tức gửi đi đến người dùng sao cho lượng Click vẫn đạt được ngưỡng cho phép", điều này có thể tăng trải nghiệm người dùng khi người dùng vẫn nhận được 1 lượng thông báo ít nhất (tránh gây phiền phức khi gửi quá nhiều tin đến) nhưng vẫn đảm bảo lượng tổng số lượng Click. Bài toán tối ưu lúc này có thể được mô hình hoá ở công thức dưới đây:

$$\text{minimize: } \text{TotalSends}(\mathbf{x}) \quad (20)$$

$$\text{s.t. } \text{TotalClicks}(\mathbf{x}) \geq \alpha \text{TotalClicks}^*$$

$$\text{TotalSends}(\mathbf{x}) \leq \beta \text{TotalSends}^*$$

$$\mathbf{x}_i \geq n_{\min}$$

$$\mathbf{x}_i \leq n_{\max}$$

Trong đó TotalClicks^* và TotalSends^* lần lượt là tổng số lượt Click vào thông báo và Tổng số thông báo đã gửi đi của tập người dùng $\mathbf{x}$ tại phiên $t - 1$. $\mathbf{x}_i$ chính là ngân sách thông báo ước lượng gửi cho người dùng i tại phiên thứ t . $n_{\min}, n_{\max}$ là lượng thông báo tối thiểu và tối

đa của một người dùng nhận được tại phiên t . α và $\beta \in [0,1]$, ở đây chúng tôi muốn đảm bảo rằng quá trình tối ưu, số lượng Click không bị giảm quá $(1 - \alpha)$ và phần trăm số lượng thông báo phải giảm ít nhất β . Đây cũng là hai tham số giúp cân bằng sự đánh đổi khi giảm số lượng tin tức thông báo xuống. Phần tiếp theo tác giả sẽ trình bày thực nghiệm và phân tích kết quả thu được.

IV. KẾT QUẢ THỰC NGHIỆM VÀ ĐÁNH GIÁ

A. Cài đặt và dữ liệu

Lotus là một trang mạng xã hội mới của Việt Nam với cách tiếp cận nội dung. Hiện tại hằng ngày có khoảng hơn 20 nghìn người dùng hoạt động. Hệ thống thông báo của Lotus hoạt động dựa trên việc gửi những tin tức mới nhất và liên quan đến người dùng hằng ngày. Việc gửi thông báo cho người dùng giúp tăng trải nghiệm, khuyến khích người dùng vào hệ thống. Tôi triển khai hệ thống chạy thử nghiệm cho 2 nghìn người dùng sử dụng online trên ứng dụng mạng xã hội Lotus trong hai tuần. Dữ liệu lịch sử về Số lượng thông báo được gửi và Số lượng ấn vào tin tức Click của những người dùng này được thu thập trong 1 tuần trước đó để sử dụng cho quá trình tối ưu. Cụ thể dữ liệu thu thập về bao gồm những thông tin sau:

- *UserID*: Định danh người dùng
- *Time*: Thời gian
- *ActionID*: Trạng thái hoạt động bao gồm: Nhận thông báo, Xem thông báo, Click thông báo)
- *PostID*: Mã bài viết được gửi tới cho người dùng

Từ những dữ liệu này, tôi tiền xử lý tạo ra được dữ liệu được mô tả như sau:

- *Time*: Thời gian (theo ngày)
- *UserID*: Định danh người dùng
- n_{push} : Số thông báo nhận được
- n_{click} : Số thông báo đã được Click

Tôi cũng sử dụng thư viện CVXOPT trên ngôn ngữ lập trình python để giải quyết bài toán LP. Tham số được tôi sử dụng được ghi chi tiết tại Bảng II.

Bảng II: Tham số thực nghiệm

Tham số	Giá trị
α	0.98
β	1
$n_{\min}$ / ngày	10
$n_{\max}$ / ngày	30

B. Kết quả và đánh giá

Bảng III đưa ra kết quả theo từng tuần thực hiện quá trình kiểm nghiệm. Như kết quả ta có thể thấy rằng Số lượng tin gửi cho 2000 người dùng 1 tuần trước khi thực hiện tối ưu vào khoảng gần 48,5 nghìn tin tức trong 1 ngày khoảng 340 nghìn tin tức trong 1 tuần. Sau khi thực hiện thử nghiệm cho tuần đầu tiên, số tin gửi đi giảm khoảng 25% so với trước đó, CTR tăng từ 5.8% lên 6.56%. Tuần tiếp theo CTR đạt 7.63%. Điều này chứng minh lượng thông báo đã được giảm đi đáng kể nhưng vẫn đảm bảo được chỉ số CTR.

Bảng III: Tổng quát các chỉ số theo các giai đoạn, trong đó Send, Click, CTR lần lượt ứng với số lượng tin được gửi, Số lượng tin được Click và Tỷ lệ Click

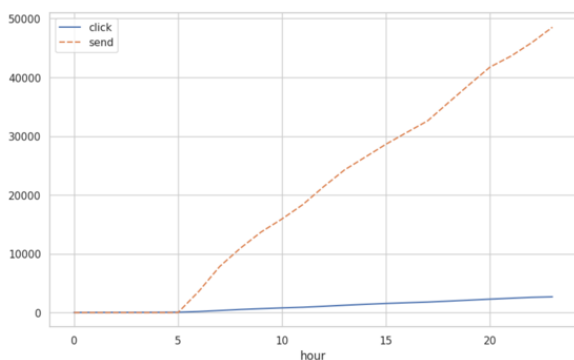
Session	Send (tin/ngày)	Click (tin/ngày)	CTR
Trước thử nghiệm	48518	2686	5.8%
Thử nghiệm 1 tuần	36542	2317	6.56%
Thử nghiệm 2 tuần	29826	2203	7.63%

Bảng IV thể hiện các chỉ số TB trên từng người dùng. Tổng số Click trung bình nhận từ 170 tin/tuần trước khi thử nghiệm giảm xuống khoảng 105 tin/tuần sau hai tuần thử nghiệm. CTR tăng từ 3.05% lên khoảng 3.53%.

Bảng IV: Bảng chỉ số đánh giá trên từng người dùng

Session	Send (tin/ngày)	Click (tin/ngày)	CTR
Trước thử nghiệm	171	5.85	3.05%
Thử nghiệm 1 tuần	129	4.74	3.25%
Thử nghiệm 2 tuần	105	4,30	7.00%

Tiếp theo, chúng tôi có thống kê các chỉ số trên cho toàn bộ người dùng Lotus tại Bảng V. Tại tuần thứ 2, tuy số lượng tin tức được gửi đi tăng khoảng 10% so với tuần 1 nhưng tổng số lượng ấn vào tin tức (Click) vẫn giảm, và chỉ số CTR giảm. Trong khi đó nhóm 2 nghìn người dùng sử dụng tối ưu số lượng tin gửi đi có chỉ số CTR tăng. Mặt khác Hình 2, thể hiện số lượng tin tức trung bình gửi trong 1 ngày của 2 nghìn người dùng tại thời điểm trước thử nghiệm để đạt số lượng click như thời điểm thử nghiệm được 1 tuần. Như Hình 2 chỉ ra rằng 2 nghìn người dùng trung bình 1 ngày cần nhận 42 nghìn tin để đạt được số Click trung bình 1 ngày bằng giai đoạn 1 (Tổng số tin Trung bình gửi tại giai đoạn 1 là 36,5 nghìn tin/ngày để đạt được khoảng 2,3 nghìn Click - Bảng III).



Hình 2: Biểu đồ Click của nhóm user trước thử nghiệm

Bảng V: Chỉ số chung của toàn bộ user trên Lotus với Send, Click, CTR lần lượt ứng với Số lượng tin được gửi, Số lượng tin được Click và Tỷ lệ Click

Session	Send	Click	CTR
Trước thử nghiệm	16284327	28322	0.18%
Thử nghiệm 1 tuần	14498727	27987	0.20%
Thử nghiệm 2 tuần	16088124	26528	0.18%

Mặc dù từ kết quả của thực nghiệm cho thấy việc giảm số lượng tin gửi đến người dùng tối ưu theo chỉ số Click lịch sử người dùng giúp cân đối được ngân sách hợp lý đến từng người dùng. Tuy nhiên, số lượng Click phiên tiếp theo vẫn giảm. Vấn đề này có thể do hệ thống chỉ giảm số lượng tin gửi trên toàn cục mà chưa giải quyết được vấn đề "Bài nào không nên gửi?". Hơn nữa thực nghiệm vẫn chưa thực hiện được trên nhiều tham số để xác định được đâu là ngưỡng tối thiểu n_{min} và tối đa n_{max} cho từng người dùng. Hướng nghiên cứu sắp tới có thể giải quyết vấn đề tối ưu cục bộ tới từng người dùng. Không chỉ tối ưu số lượng ngân sách cho người dùng trên tổng các thể loại của bài viết, mà ta có thể tối ưu số lượng theo từng thể loại của bài viết.

V. KẾT LUẬN

Nghiên cứu này tập trung tìm hiểu về bài toán Tối ưu ràng buộc và áp dụng LP trong tối ưu số lượng thông báo trong hệ thống đề xuất tin tức. Nghiên cứu cho thấy, việc sử dụng tương tác Click như một ràng buộc cho số lượng thông báo gửi đi là có ý nghĩa, giúp cho hệ thống điều hướng đúng số lượng thông báo cần thiết đến tập người dùng mà không làm phiền người dùng. Nghiên cứu chỉ ra, số lượng tin tức gửi đi đã giảm đáng kể và đảm bảo được tỷ lệ CTR của hệ thống. Tuy nhiên, nghiên cứu này chỉ áp dụng ở mức đơn giản, vẫn chưa giải quyết mức sâu hơn đến từng thể loại của bài viết, dẫn đến đánh đổi đến số lượng Click bị giảm đi. Hướng giải quyết tương lai, tôi có thể áp dụng tối ưu đa mục tiêu và hàm đối ngẫu để tiếp cận sâu hơn nữa, cá nhân hóa đến từng người dùng.

LỜI CẢM ƠN

Trong nghiên cứu này, tôi cảm ơn team DataMining cũng như công ty VCCorp đã hỗ trợ tôi về mặt cơ sở vật chất cũng như dữ liệu thực tế để thực hiện nghiên cứu này.

TÀI LIỆU THAM KHẢO

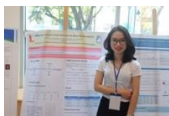
- [1] D. Agarwal, B.-C. Chen, and P. Elango, "Spatio-temporal models for estimating click-through rate." in WWW, J. Quemada, G. León, Y. S. Maarek, and W. Nejdl, Eds. ACM, 2009, pp. 21–30. [Online]. Available: <http://dblp.uni-trier.de/db/conf/www/www2009.html#AgarwalCE09>
- [2] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation." in WWW, M. Rappa, P. Jones, J. Freire, and S. Chakrabarti, Eds. ACM, 2010, pp. 661–670. [Online]. Available: <http://dblp.uni-trier.de/db/conf/www/www2010.html#LiCLS10>
- [3] J. Langford and T. Zhang, "The epoch-greedy algorithm for multi-armed bandits with side information," in Advances in Neural Information Processing Systems, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds., vol. 20. Curran Associates, Inc., 2007.
- [4] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," Machine learning, vol. 47, no. 2, pp. 235–256, 2002.
- [5] D. Agarwal, B.-C. Chen, P. Elango, and X. Wang, "Click shaping to optimize multiple objectives," 08 2011, pp. 132–140.
- [6] A. Deepak, B.-C. Chen, P. Elango, and X. Wang, "Personalized click shaping through lagrangian duality for online recommendation," in Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2012, pp. 485–494.
- [7] D. Agarwal, S. Chatterjee, Y. Yang, and L. Zhang, "Constrained optimization for home-

- page relevance,” in WWW (Companion Volume), 2015, pp. 375–384. [Online]. Available: <https://doi.org/10.1145/2740908.2745398>
- [8] R. Gupta, G. Liang, and R. Rosales, “Optimizing email volume for sitewide engagement,” 11 2017, pp. 1947–1955.
- [9] T. H. Vu, Machine Learning cơ bản, 2018. [Online]. Available: <https://github.com/tiepvupsu/ebookMLCB>
- [10] S. Boyd and L. Vandenberghe, Convex Optimization. Cambridge University Press, March 2004.

NOTIFICATION VOLUME OPTIMIZATION IN RECOMMENDATION SYSTEM USING LINEAR PROGRAMING

Abstract: The notification function is a crucial component of recommendation systems, as it can enhance user engagement and sustain activity. However, excessive notifications can irritate users and even lead to account deletions. This paper addresses the challenge of optimizing notification frequency in a news recommendation system. Each user is allocated a notification budget for a specific period. By applying optimal constraints, we aim to balance notification frequency with user engagement metrics, such as click-through rates (CTR). Our study, conducted with 2000 Lotus system users, demonstrates that while significantly reducing the number of notifications, the CTR remains strong and even surpasses that of other user groups within the system.

Keywords: Recommendation system, optimal constraints, notification



Nguyễn Thị Trang, nhận bằng kỹ sư ngành hệ thống thông tin và thạc sĩ ngành khoa học dữ liệu tại Đại Học Bách Khoa Hà Nội lần lượt năm 2019 và 2020. Hiện tại, cô đang đang làm nghiên cứu và giảng dạy tại Học viện Công nghệ Bưu Chính Viễn Thông. Hướng nghiên cứu hiện tại là: Trí tuệ nhân tạo, Hệ khuyến nghị, Xử lý ngôn ngữ tự nhiên...

Email: trangnt@ptit.edu.vn