ĐỀ XUẤT THUẬT TOÁN KHUYẾN NGHỊ THEO PHÂN BỐ DỰA TRÊN MÔ HÌNH HỖN HỢP GAUSSIAN

Nguyễn Văn Đạt*, **Tạ Minh Thanh**‡†
Sun Inc., 13F Keangnam 72 Tower, Plot E6, Phạm Hùng, Nam Từ Liêm, Hà Nội
†Học Viện Kỹ Thuật Quân Sự, 239 Hoàng Quốc Việt, Cầu Giấy, Hà Nội

***Tóm tắt*—Ngày nay, các hệ thống khuyến nghị được tích hợp vào hầu hết các trang thương mai điện tử giúp tăng cường năng suất bán hàng cho các doanh nghiệp bằng cách hỗ trợ người tiêu dùng tìm được những sản phẩm phù hợp, chất lượng nhất. Hiện nay, có khá nhiều thuật toán khuyến nghị tốt và hiệu quả, tuy nhiên, thuật toán content-based recommendation vẫn là thuật toán phổ biến nhất được sử dụng trong giai đoạn đầu của các dự án. Trong một số trường hợp, độ chính xác của kết quả từ thuật toán content-based vẫn là một điều lo ngại khi bài toán liên quan đến độ tương tự về phân phối giữa các thành phần. Thêm nữa, các phương pháp để đo độ tương đồng cũng là một vấn đề quan trọng ảnh hưởng đến độ chính xác của các thuật toán content-based trong các bài toán về độ tương đồng giữa các phân phối. Để giải quyết hai vấn đề này, chúng tôi đề xuất một thuật toán content-based mới dựa trên mô hình hỗn hợp gaussian giúp tăng độ chính xác cho kết quả đầu ra. Mô hình đề xuất được thực nghiệm trên một bộ dữ liệu về rượu bao gồm 6 chỉ số về mùi vị, dữ liệu tag mô tả về vị của rượu và một số trường thông tin khác. Thuật toán này sẽ gom n bản ghi dựa trên n vectors 6 chiều thành k nhóm $(k < n)$ trước khi áp dụng một công thức để sắp xếp các kết quả trả về. So sánh kết quả mô hình đề xuất với 2 thuật toán phổ biến khác trên bộ dữ liệu trên, kết quả thực nghiệm thu được không chỉ đạt được độ chính xác tốt hơn, mà thời gian thực thi của mô hình cũng vượt qua điều kiện cho việc áp dụng vào các ứng dụng thực tế.**

***Từ khóa*—Hệ thống khuyến nghị, Content-Based, mô hình hỗn hợp gaussian, hệ thống khuyến nghị phân phối, Gaussian Mixture Model - GMM, GaussianFilter Function, Collaborative Filtering.**

Tác giả liên hệ: Nguyễn Văn Đạt, Email: nguyen.van.dat@sun-asterisk.com.

Đến tòa soạn: 04/2020, chỉnh sửa: 7/2020, chấp nhận đăng: 07/2020.

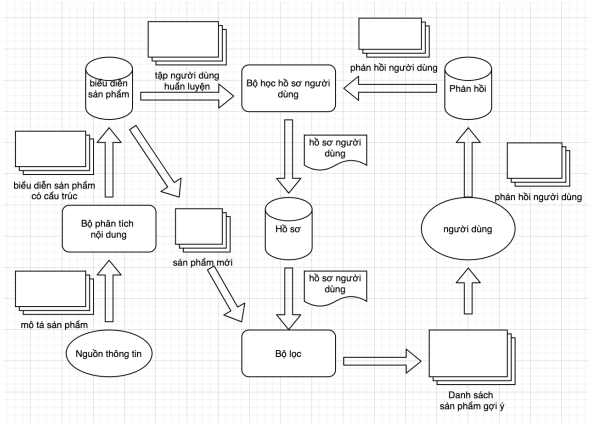
‡ Corresponding author

I. MỞ ĐẦU

A. Tổng quan

Với sự phổ biến của mạng Internet trong những năm gần đây, công nghệ đã mang lại những cơ hội rất lớn phục vụ tự động hoá đến cuộc sống của con người. Mặt khác, sự đa dạng và dư thừa thông tin, nội dung trên các website, thư viện số là yếu tố dẫn đến sự ngày càng khó khăn trong việc tìm kiếm thông tin thực sự cần thiết cho mỗi nhu cầu cá nhân [7, 11, 2]. Hệ thống khuyến nghị (Recommendation systems) là một giải pháp hiệu quả để giải quyết vấn đề này mà không cần người dùng cung cấp các yêu cầu cụ thể [31, 33]. Thay vào đó, các hệ thống khuyến nghị có thể phân tích nội dung các thuộc tính của các sản phẩm, đối tượng để có thể tự động gợi ý ra những thông tin làm hài lòng những nhu cầu và sở thích của người dùng [17, 15]. Kiến trúc chung cho các thuật toán content-based (CB) được hiển thị trong hình 1. Ngày nay, làm thế nào để xây dựng và thiết kế một thuật toán khuyến nghị đã trở thành các chủ đề tập trung cần được nghiên cứu.

Thuật toán content-base trong hệ thống khuyến nghị được sử dụng rộng rãi bởi tính đơn giản và hiệu quả của nó trong thời kỳ đầu của bất kỳ dự án nào. Theo Pasquale Lops $et.\ al.$ [14] trong Chương 3 “Content-based recommendation system: State of the Art and Trends" đã nhấn mạng rằng có rất nhiều lợi ích thu được từ các thuật toán content-based so với các thuật toán cùng loại là Collaborative Filtering (CF) như là: tính độc lập giữa người dùng, minh bạch, vấn



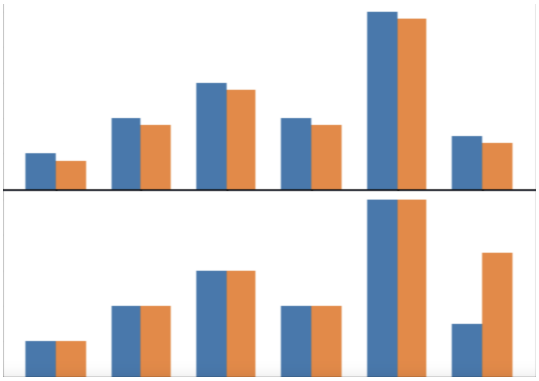
Hình 1: Cấu trúc hệ thống khuyến nghị dựa trên thuật toán content-based

đề cold-start khi thêm một sản phẩm mới. Bên cạnh đó, thuật toán này vẫn còn tồn tai những mặt hạn chế như: giới hạn về mặt nội dung cho việc phân tích, tính chuyên môn hoá, thiếu dữ liệu đánh giá từ người dùng, hay thiếu độ chính xác cần thiết cho một vài bài toán đặc biệt. Hangyu *et. al.* [29] đã sử dụng Gaussian mixture model (GMM) cho thuật toán khuyến nghị CF để giải quyết vấn đề thưa thớt dữ liệu đánh giá từ phía người dùng. Chen *et. al.* [4] đã đề xuất một mô hình lai kết hợp giữa GMM với thuật toán khuyến nghị item-based CF để dự đoán ra dữ liệu đánh giá của người dùng cho các sản phẩm giúp làm tăng độ chính xác trên các hệ thống khuyến nghị. Rui Chen *et. al.* [3] tận dụng GMM với ma trận tăng cường factorization giúp làm giảm đi tác động tiêu cực của dữ liệu rời rạc nhiều chiều. Trong ngữ cảnh của các bài toán gợi ý bài hát, Yoshii *et. al.* [30] đã đề xuất một hệ thống khuyến nghị lai, bằng việc kết hợp CF sử dụng dữ liệu người dùng đánh giá và các giá trị thuộc tính content-based được mô hình hoá bằng GMM dựa trên MFCCs (Mel-frequency Cepstral Coefficients) qua việc tận dụng một mạng Bayesian. Tuy nhiên, có một điểm cần lưu ý đó là, các hệ thống lai hoặc hệ thống CF yêu cầu lịch sử hành vi của người dùng để hoạt động hiệu quả, điều mà các hệ thống CB có thể giải quyết mà không cần đến các dữ liệu kiểu này. Thêm nữa, các thuật toán CB dựa trên phân phối các thuộc tính của các sản phẩm vẫn chưa được giải quyết. Một ví dụ điển hình của việc sử dụng CB giúp tự động tìm kiếm ra các sản phẩm tương đồng dựa trên phân phối và khoảng cách được hiển thị ở Hình 2, đây là hai biểu đồ cột so sánh sự khác nhau giữa 6 thuộc tính của hai sản phẩm (1 sản phẩm gốc (màu lam), 1 sản phẩm được gợi ý (màu cam)) được trả về bằng việc sử dụng công thức sắp xếp khoảng cách (biểu đồ trên) và phân phối (biểu đồ dưới). Các bài toán khuyến nghị dựa trên xác xuất phân phối của các thuộc tính không thể được giải quyết bằng những phương pháp thông thường.

Một điều khó khăn nữa, các nội dung mô tả sản phẩm đôi khi là không đáng tin cậy, không đầy đủ, không chính xác gây ra giảm độ chính xác trong các bài toán CB [11].

B. Đóng góp chính của bài báo

Để giải quyết hai vấn đề nêu trên, chúng tôi đề xuất một phương pháp tiếp cận mới sử dụng GMM [25] để gom nhóm tất cả các sản phẩm thành các nhóm khác nhau, sau đó, áp dụng một công thức bộ lọc Gaussian tính trọng số (Gaussian Filter Function - GFF) như một phương pháp tính độ tương đồng để sắp xếp kết quả trả về. Để chứng minh tính hiệu quả của mô hình, chúng tôi thực nghiệm và so sánh với 2 phương pháp CB phổ biến khác, Bag of Word (BOW)[1] với GFF (BOW + GFF), và GMM với Euclidean Distance (ED) [13] (GMM + ED). Mô hình đề xuất của chúng tôi cố gắng thực hiện một hệ thống sử dụng chủ yếu những tính chất đặc trưng của sản phẩm của các trang thương mại điện tử để đưa ra được hệ thống khuyến nghị với độ chính xác cao và hiệu năng tốt nhất. Dựa vào kết quả thực nghiệm, chúng tôi có thể kết luận rằng, mô hình của chúng tôi không chỉ tốt hơn hẳn về độ chính xác, mà còn đạt tốc



Hình 2: Ví dụ giữa việc sử dụng công thức khoảng cách và phân phối cho hệ thống khuyến nghị dựa trên phân phối thuộc tính

độ trả về kết quả nhanh hơn so với hai mô hình đã được đề xuất trước đó.

C. Cấu trúc bài báo

Trong bài báo này, chúng tôi tổ chức nội dung như sau. Các kiến thức liên quan được trình bày trong mục 2. Trong mục 3, kiến trúc và chi tiết mô tả về mô hình được đưa ra. Thí nghiệm và đánh giá được trình bày trong phần 4. Kết luận, dự kiến nội dung cải thiện và xu hướng nghiên cứu được đưa ra trong mục 5.

II. Các kỹ thuật liên quan

Trong mục này, chúng tôi sẽ trình bày một số kỹ thuật liên quan cần được sử dụng trong bài báo. Chi tiết các phần sẽ được trình bày dưới đây:

A. Hệ thống khuyến nghị Content-Based

Đây là một trong những thuật toán khuyến nghị phổ biến và thông dụng nhất. Thuật toán dựa trên ý tưởng bằng việc sử dụng các mô tả đặc trưng các thuộc tính của các sản phẩm cho mục đính khuyến nghị. Bài toán khuyến nghị này có thể được chia ra làm 2 nhánh chính: Chỉ phân tích các thuộc tính của sản phẩm, hoặc xây dựng các bộ hồ sơ người dùng cho từng cá nhân dựa trên các đặc tính và dữ liệu đánh giá của sản phẩm.

1) Hệ thống khuyến nghị dựa trên phân tích thuộc tính của sản phẩm: Với trường hợp dữ liệu là dữ liệu thô, thuần khiết về các thuộc tính sản phẩm, và không có tính cá nhân hoá, chúng ta có thể xây dựng một hệ thống khuyến nghị dựa trên sự tương đồng giữa các thuộc tính này. Ví dụ, chúng ta có N bản ghi $X_n = \{x_1, x_2, ..., x_n\}$ với x_i có h thuộc tính $x_i = \{p_1, p_2, ..., p_h\}$; trong đó p_i có thể phản ánh bất kỳ một giá trị nào đó đó ngoài đời thực, chẳng hạn như: giá cả, thẻ tags, nội dung miêu tả, nhãn hiệu... Tư tưởng chính ở đây là cố gắng tìm ra các sản phẩm có những vùng nội dung giống nhau nhiều nhất có thể để nhóm chúng thành một nhóm các sản phẩm tương đồng.

2) Xây dựng hồ sơ người dùng dựa trên thuộc tính sản phẩm: Trong trường hợp này, chúng ta giả sử có C người dùng $U_n = \{u_1, u_2, ..., u_c\}$, n sản phẩm $X_n = \{x_1, x_2, ..., x_n\}$, và dữ liệu đánh giá trên một vài sản phẩm của từng người dùng. Tư tưởng chính ở đây là tận dụng dữ liệu đánh giá rời rạc của các người dùng để dự đoán một số sản phẩm có khả năng nhất phù hợp với từng cá nhân, và nó mang tính cá nhân hoá. Hệ thống sẽ phân tích một bộ các sản phẩm đã được đánh gía bởi người dùng và dựa vào đó để xây dựng lên một bộ hồ sơ về sở thích cho từng người dùng. Bộ hồ sơ này được biểu diễn dưới dạng dữ liệu có cấu trúc về sự quan tâm của người dùng trên toàn bộ tập sản phẩm. Về cơ bản cách hoạt động, hệ thống sẽ dựa trên các bộ hồ sơ này và thuộc tính của từng sản phẩm để đưa ra những dự đoán đánh giá cho các sản phẩm chưa được người dùng xem đến hoặc đánh giá. Và từ đó, dựa vào giá các giá trị đánh giá này để trả về cho người dùng những sản phẩm mà họ có thể quan tâm nhất.

B. Độ đo sự tương đồng

Trong thuật toán content-based, công thức tính toán mức độ tương đồng trực tiếp ảnh hưởng đến độ chính xác của kết quả đầu ra. Một số công thức phổ biến sẽ được liệt kê dưới đây:

`euclidean distance`: đây là một trong những công thức phổ biến nhất dùng để đo độ tương đồng giữa 2 vectors bằng việc tính toán căn bậc hai tổng bình phương khoảng cách của từng phần tử tương ứng trong vector:

$$d(p,q) = \sqrt{(p_1-q_1)^2 + (p_2-q_2)^2 + ... + (p_n-q_n)^2} = \sqrt{\sum_i^n (p_i - q_i)^2} \quad (1)$$

Trong đó p_i, q_i là 2 vectors tương ứng biểu diễn các thuộc tính của 2 sản phẩm p_i, q_i dưới dạng số.

`Cosin`: Công thức đo độ tương đồng giữa hai vectors bằng việc tính toán cosine góc giữa 2 vectors này [21].

$$cosin(\vec{i}, \vec{j}) = \frac{\vec{i}.\vec{j}}{\mid \vec{i} \mid . \mid \vec{j} \mid} \quad (2)$$

Giá trị của thước đo này được trả về trong khoảng [-1, 1], trong đó i, j là 2 vectors tương ứng biểu diễn 2 sản phẩm khác nhau.

`Pearson`: Hệ số tương quan pearson phản ánh mức độ tương quan tuyến tính giữa 2 vectors [26], được định nghĩa như sau:

$$p(i,j) = \frac{\sum_{r\in\ i,j(R_i,r-\overline{R_i})(R_j,r-\overline{R_j})}}{\sqrt{\sum_{r\in\ i,j(R_i,r-\overline{R_i})^2}}\sqrt{\sum_{r\in\ i,j(R_j,r-\overline{R_j})^2}}} \quad (3)$$

Giá trị của $p(i,j)$ trả về sẽ nằm trong khoảng [-1, 1], trong đó r là phần giao nhau giữa các phần khác nhau giữa 2 vectors i,j. $\overline{R_i}$, $\overline{R_j}$ là trung bình giá trị của 2 vectors i,j.

`Jaccard`: Độ tương đồng Jaccard thường được sử dụng để đo độ tương đồng và khác nhau giữa 2 tập mẫu hữu hạn [19], được định nghĩa như sau:

$$J(A,B) = \frac{\mid A \cap B \mid}{\mid A \mid + \mid B \mid - \mid A \cap B \mid} \quad (4)$$

Trong đó, A và B là 2 tập mẫu khác nhau.

C. Mô hình hỗn hợp Gaussian (GMM)

GMM là một hàm số được tổng hợp từ rất nhiều bộ Gaussians, được sử dụng để giải quyết các bài toán liên quan đến dữ liệu ở cùng một tập chứa các phân phối khác nhau [24, 5], mỗi phân phối được định nghĩa bởi $k \in \{1..K\}$, trong đó K là số cụm của bộ dữ liệu. Mỗi Gaussian k trong hỗn hợp này được tổng hợp từ các tham số sau:

(i) Giá trị trung bình μ định nghĩa trung tâm của cụm.

(ii) Hiệp phương sai Σ định nghĩa biên của cụm.

(iii) Giá trị xác suất α định nghĩa mức độ lớn hay nhỏ của hàm Gaussian.

GMM được định nghĩa như sau:

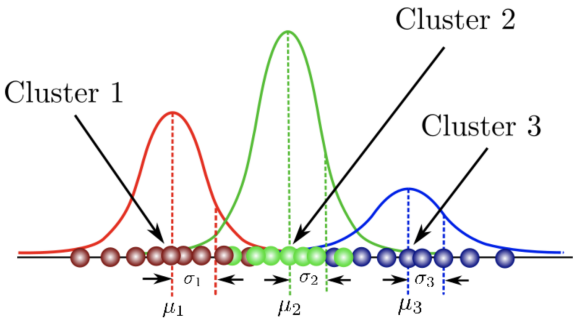
$$p(x) = \sum_{i=1}^{k} \alpha_i.N(x|\mu_i,\Sigma_i), \quad (5)$$

trong đó, $N(x|\mu_i,\Sigma_i)$ là thành phần thứ i của mô hình lai này, là hàm mật độ xác suất của vector x có n chiều tuân theo phân phối Gaussian và có thể được định nghĩa như sau:

$$N(x) = \frac{1}{(2\pi)^{\frac{n}{2}} \mid \Sigma \mid^{\frac{1}{2}}} \epsilon^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}, \quad (6)$$

và

$$\sum_{i=1}^{k} \alpha_i = 1 \quad (7)$$



Hình 3: Mô hình hỗn hợp Gaussian

Giả sử rằng một tập mẫu $D = \{x_1, x_2, x_3, ..., x_m\}$ tuân theo phân phối hỗn hợp Gaussian, chúng ta có thể sử dụng một biến ngẫu nhiên $z_j \in \{1, 2, ..., k\}$ để biểu diễn thành phần hỗn hợp của mẫu x_j, ở đó các giá trị của nó là không xác định. Ngoài ra, có thể nhận ra rằng xác suất trước $P(z_j = i)$ của z_j tương ứng với $\alpha_i (i = 1, 2, 3, ..., k)$. Theo lý thuyết Bayes [12], chúng ta có thể thu được xác suất sau của z_j được định nghĩa như sau:

$$p(z_j - i|x_j) = \frac{P(z_j = i).p(x_j|z_j = i)}{p(x_j)}$$

$$= \frac{\alpha_i.N(x_j|\mu_i,\Sigma_i)}{\sum_{l=1}^{k} \alpha_l.N(x_j|\mu_l,\Sigma_l)} \quad (8)$$

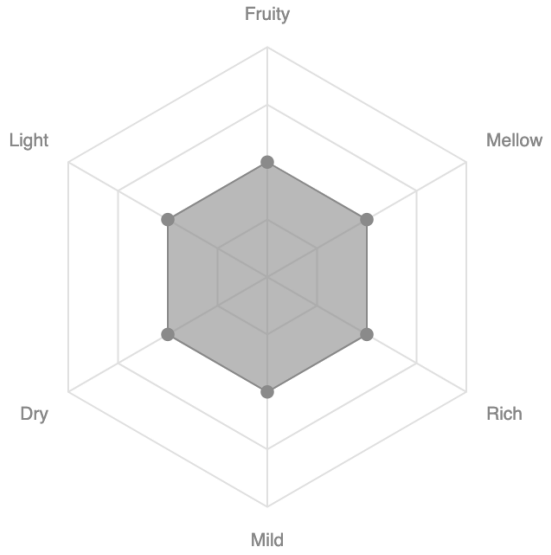
Trong công thức trên, $p(z_j = i|x_j)$ biểu diễn xác suất sau của mẫu x_j được sinh ra từ thành phần hỗn hợp Gaussian thứ i. Giả sử $\gamma_{ij} = \{1, 2, 3, ..., k\}$ biểu diễn $p(z_j = i|x_j)$. Khi tham số mô hình $\{(\alpha_i, \mu_i, \Sigma_i)|1 \leq i \leq k\}$ trong công thức trên được tìm ra, các cụm của mô hình hỗn hợp Gaussian chia mẫu D thành k cụm $C = \{C_1, C_2, ..., C_k\}$ [24], và nhãn cụm λ_j của mỗi mẫu x_j có thể được định nghĩa theo công thức sau:

$$\lambda_j = \arg max_{i\in 1,2,3,...,k} \gamma_{ji} \quad (9)$$

Dựa vào công thức, chúng ta có thể đưa x_j vào cụm C_{λ_j}. Tham số mô hình này $\{(\alpha_i, \mu_i, \Sigma_i)|1 \leq i \leq k\}$ được giải quyết bởi thuật toán EM [16].

D. Tập dữ liệu

Mô hình đề xuất của chúng tôi được thực hiện trên một bộ dữ liệu về rượu, cụ thể hơn, về rượu



Hình 4: Trực quan hoá 6 chỉ số mùi vị

sake là một trong những loại rượu nổi tiếng nhất của Nhật Bản. Tập dữ liệu này được thu thập từ bộ dữ liệu của Sakenowa[1]. Đây là một trang web nổi tiếng và uy tín chuyên bán rượu sake tại xứ sở hoa anh đào. Các thí nghiệm trong thuật toán đề xuất và thuật toán so sánh đều được tiến hành thử nghiệm trên bộ dữ liệu Sakenowa này và so sánh với kết quả thực tế mà trang thương mại rượu Sakenowa đang sử dụng để khuyến nghị các loại rượu cho khách hàng.

Bộ dataset này tổng cộng chứa 1072 bản ghi được đặc trưng bởi 19 thuộc tính như tên rượu, thương hiệu rượu, năm sản xuất, ảnh rượu, tags về mùi vị của rượu, 6 chỉ số về rượu $(f_1, f_2, ..., f_6)$ biểu diễn cho fruity, mellow, rich, mild, dry và light, chúng tôi để nguyên văn bản gốc tiếng anh để giữ nguyên được bản sắc và tính trừu tượng của 6 chỉ số mùi vị này)... Đáng chú ý hơn, tags về mùi vị rượu, 6 chỉ số về rượu đóng vai trò quan trọng hơn các thuộc tính khác.

Khoảng giá trị của 6 chỉ số $(f_1, \cdots, f_6)$ trong khoảng [0, 1], trong đó phần lớn giá trị thuộc [0.2, 0.6]. Giá trị các trường văn bản trong bộ dữ liệu này đều là ngôn ngữ nhật. Nhiệm vụ của thuật toán khuyến nghị là bằng cách nào đó tự động trả về những loại rượu tương đồng, giống nhất có thể với một loại rượu mà người dùng đang xem. Hình 4 mô phỏng về 6 chỉ số rượu đang được dùng làm đặc trưng chính trong trang Sakenowa.

[1]https://sakenowa.com

Tuy nhiên, đây thực sự là một bộ dataset khó, do thiếu dữ liệu hoặc dữ liệu không đồng đều dẫn đến sự rời rạc trong dữ liệu, đặc biệt là trong 6 chỉ số $f_1, ..., f_6$. Vì vậy, nhiệm vụ của chúng ta trở nên khó khăn hơn, ảnh hưởng trực tiếp đến kết quả hệ thống. Cụ thể hơn, hơn 30% các giá trị trường 6 chỉ số là rỗng, 2% không tồn tại chỉ số mùi vị tags. Thêm nữa, rất nhiều giá trị tags là không chính xác, không tin cậy cần được tiền xử lý và xoá bỏ nhiễu. Dưới đây là bảng thống kê các giá trị rỗng trong bộ dữ liệu, một số trường không được liệt kê trong bảng này.

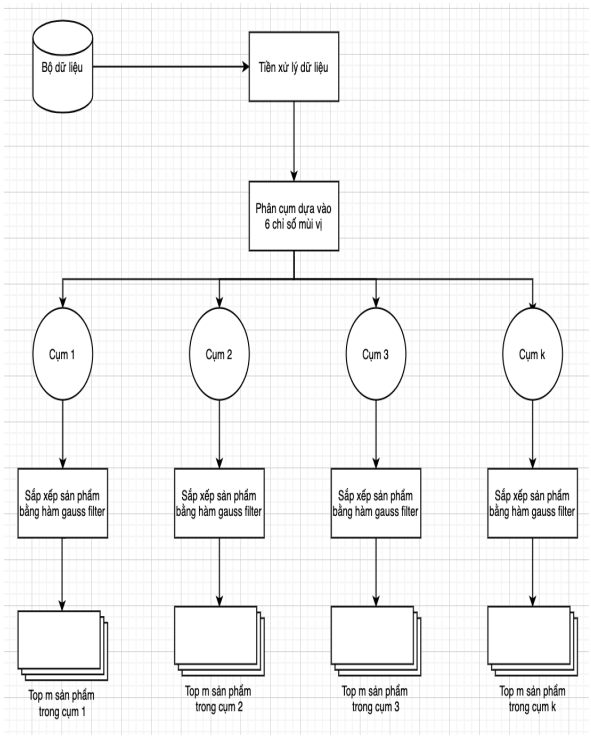
III. Mô hình đề xuất thuật toán

Trong phần này, chúng tôi sẽ giới thiệu và giải thích chi tiết hơn về mô hình đề xuất. Như đã đề cập ở phần trước, nhiệm vụ của chúng ta là phải trả về những sản phẩm rượu giống nhất có thể với một sản phẩm mà khách hàng đang xem, dựa vào 19 thuộc tính của sản phẩm. Đặc biệt hơn, 6 chỉ số mùi vị và tags mùi vị là những tác nhân chính ảnh hưởng trực tiếp đến kết quả cả về độ chính xác và tính giác quan. Vì vậy, chúng tôi chỉ chọn 6 chỉ số mùi vị và tags mùi vị để cho kết quả tốt hơn. Càng giống nhau về 6 chỉ số này, kết quả gợi ý cho người dùng càng chính xác. Dựa trên nguyên lý này, chúng tôi đề xuất một phương pháp tận dụng sự tương đồng trong phân phối của dữ liệu để tăng cường độ chính xác của kết quả trả về.

Trong đề xuất của chúng tôi, thay vì sử dụng các vector 6 chiều để tính toán độ tương đồng bằng các công thức cosine hay euclidean, chúng tôi đầu tiên sử dụng GMM để nhóm tất cả các bản ghi thành $K = \{1, 2, 3, ..., k\}$ nhóm, sau đó sắp xếp kết quả ở mỗi nhóm với từng bản ghi ở từng nhóm. Để sắp xếp các kết quả này, như đã đề cập chúng ta hoàn toàn có thể sử dụng các công thức tính độ tương đồng phổ biến như cosine hoặc euclidean, tuy nhiên, để thu được kết quả tốt hơn, chúng tôi sẽ sử dụng một công thức tính trọng số giữa các phân phối của 2 vector tuân theo phân

Bảng I: Bảng thống kê các trường dữ liệu rỗng

$f_{1..6}$	**Tags mùi vị**	**Tên sản phẩm**
Số thực	***Kiểu chuỗi***	***Kiểu chuỗi***
30.4 %	1.77 %	13.4 %



Hình 5: Mô hình hoạt động thuật toán

phối Gaussian. Mô hình hoạt động của mô hình thuật toán được hiển thị ở Hình 5.

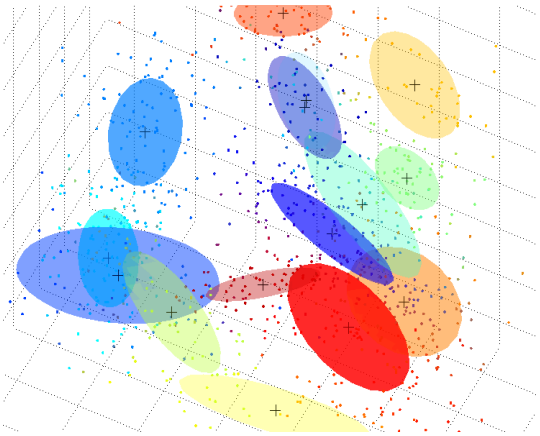
A. Tiền xử lý dữ liệu

Thực tế chỉ ra rằng, text mining là vô cùng quan trọng mọi bài toán liên quan đến văn bản, và thuật toán CB không phải là một ngoại lệ. Hơn nữa, chúng tôi chỉ chọn tags mùi vị và 6 chỉ số mùi vị như là các đặc trưng chính cho việc tính toán mức độ tương đồng giữa các sản phẩm. Trong đó, tags mùi vị là 1 tập các văn bản được viết bằng tiếng Nhật và cần được làm sạch và cấu trúc lại trước khi đưa vào mô hình tính toán.

Chúng tôi chuyển đổi 6 chỉ số mùi vị thành số thực và cần thực hiện 1 số thuật toán làm sạch và cấu trúc lại dữ liệu văn bản như tokenization, stemmings, stop word removal, tìm và thay thế từ đồng nghĩa, lemmatization,... [22, 6, 23] trước khi sử dụng. Thêm nữa, trường chỉ số tags mùi vị đã được tách thành các từ có nghĩa, vì vậy chúng tôi có thể bỏ qua bước tokenization và thực hiện các bước tiếp theo.

B. Phân cụm

Chúng tôi nhận ra rằng kết quả dự đoán cuối cùng phụ thuộc rất lớn vào 6 chỉ số mùi vị của rượu. Theo cách thông thường và phổ biến, chúng ta sẽ xây dựng 1 vector biểu diễn tất cả các thuộc tính của từng loại rượu, rồi tận dụng các phương pháp so sánh độ tương đồng như cosine hoặc euclidean để sắp xếp và trả về top kết quả. Tuy nhiên, trong một vài trường hợp, các chỉ số mùi vị tags là không đủ độ chính xác, có nhiều độ nhiễu, dẫn đến ảnh hưởng xấu đến kết quả cuối cùng. Thêm nữa, có một vấn đề không dễ dàng có thể nhận thấy được là luôn luôn có sự bù nhau giữa các thuộc tính nếu ta sử dụng các thuật toán so sánh khoảng cách để đo sự tương đồng nhau giữa các vectors. Cụ thể hơn đó là sự không đồng đều giữa 6 chỉ số mùi vị của kết quả trả về.



Hình 6: Trực quan hoá GMM

Do đó, chúng tôi quyết định nhóm tất cả các sản phẩm dựa theo phân phối của 6 chỉ số mùi vị thành các nhóm khác nhau để đảm bảo những sản phẩm có cùng phân phối 6 chỉ số này sẽ ở cùng nhóm với nhau. Nếu không thống nhất việc chọn đặc trưng của các sản phẩm để phân cụm dựa theo phân phối của thuộc tính thì sẽ dễ bị chi phối bởi rất nhiều những thông tin nhiễu, giảm độ chính xác dẫn đến khó ứng dụng trong các thuật toán khuyến nghị. Tham khảo Hình 6 mô phỏng hoá các sản phẩm sau khi được phân cụm.

C. Sắp xếp với hàm Gaussian Filter

Chúng ta có $K = \{1, 2, 3, ..., k\}$ cụm, chúng ta sẽ giả sử mỗi sản phẩm truy vấn sẽ là trung tâm của mỗi cụm mà chúng ta muốn tìm. Vì vậy, mục tiêu của chúng ta là tìm ra top m sản phẩm giống nhau nhất có thể về phân phối giữa 6 chỉ số thuộc tính, do đó, Gaussian Filter Function (GFF) là sự lựa chọn tốt hơn so với cosine hay euclidean. Công thức GFF được định nghĩa như sau:

$$G_{kl}(f_{il}, f_{jl}) = \exp - \frac{(f_{il} - f_{jl})^2}{2\sigma_{kl}^2}, \quad (10)$$

trong đó, $G_k(f_{il}, f_{jl})$ được xem xét như 1 hàm tính trọng số giữa từng cặp giá trị thứ l trong 6 chỉ số mùi vị của 2 sản phẩm khác nhau (i, j) trong cụm k, $l = \{1, 2, 3, ..., 6\}$, và σ_{kl} là độ lệch chuẩn của giá trị chỉ số f thứ l trong cụm k, công thức σ_{kl} được định nghĩa như sau:

$$\sigma_{kl} = \sqrt{\frac{\sum_{i=1}^{n_k} (f_{ilk} - \mu)^2}{n_k - 1}}, \quad (11)$$

trong đó, n_k là số lượng các sản phẩm thuộc cụm k, f_{ilk} là giá trị thứ l của f trong 6 chỉ số mùi vị của sản phẩm thứ i trong cụm k, μ là giá trị trung bình thuộc tính f_l trong nhóm k.

Chúng ta sẽ tính 6 lượt cho từng chỉ số f trong 6 chỉ số mùi vị cho mỗi cặp sản phẩm trên toàn bộ sản phẩm trong một cụm, rồi sắp xếp theo thứ tự giảm dần để tìm được kết quả tốt nhất.

D. Khoảng cách Levenshtein so sánh tags

Các tags mùi vị cũng đóng vai trò khá quan trọng ở kết quả đầu ra, vì vậy chúng ta có thể coi chỉ số này có mức độ và vai trò tương tự như 1 trong 6 chỉ số mùi vị. Để tính toán và so sánh mức độ giống nhau giữa 2 giá trị tags của 2 sản phẩm, chúng tôi sử dụng levenshtein distance (LD) để giải quyết vấn đề này [8, 32]. Công thức của levenshtein distance được định nghĩa bên dưới:

$$\text{lev}_{a,b}(i,j) = \begin{cases} max(i,j), \text{nếu } min(i,j) = 0 \\ min = \begin{cases} lev_a, b(i-1,j) + 1 \\ lev_a, b(i,j-1) + 1, \text{ngược lại} \\ lev_a, b(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} \end{cases} \quad (12)$$

E. Công thức sắp xếp cuối cùng

Kết hợp hàm tính trọng số cho 6 chỉ số mùi vị và hàm so sánh tags mùi vị bởi levenshtein distance (LD), chúng tôi thiết lập một công thức cho việc sắp xếp kết quả đầu ra như sau:

$$S(i,j) = \sum_{k=1}^{K} \sum_{l=1}^{6} G_{kl}(i,j) + lev_{tags}(i,j), \quad (13)$$

trong đó, G_{kl} là hàm tính trọng số Gaussian Filter trong công thức (11) tương ứng với chỉ số thứ l^{th} trong 6 chỉ số mùi vị rượu giữa $item_i$ và $item_j$ trong cụm k ($k = \{1, ..., K\}$ K cụm), $lev_{tags}(i,j)$ là hàm levenshtein để so sánh mức độ tương đồng giữa các chỉ số tags của hai vectors.

Chúng tôi nhận ra rằng, giá trị $S(i,j)$ giữa 2 vector càng lớn, càng có sự giống nhau giữa 2 sản phẩm được so sánh. Vì vậy, chúng tôi sắp xếp theo thứ tự giảm dần của tất cả sản phẩm trong từng cụm và trả về top m sản phẩm giống nhất với mỗi sản phẩm trong từng cụm.

F. Mô hình giả mã

Để rõ ràng hơn, chúng tôi đưa ra tiến trình xử lý thuật toán đề xuất dưới dạng giả mã để người đọc dễ dàng hiểu và hình dung hơn về toàn bộ mô hình đề xuất của chúng tôi. Hãy xem mô hình giả mã sau:

Algorithm 1: Mô hình thuật toán đề xuất

Đầu vào: Số cụm k
Đầu ra: Top m sản phẩm tương khác tự nhau của mỗi sản phẩm
Data: Bộ dữ liệu L
1. Tiền xử lý dữ liệu cho các trường văn bản
2. Xây dựng ma trận của vector 6 chiều đại diện cho 6 chỉ số mùi vị $(f_1 - f_6)$
3. Lấy ma trận này như là đầu vào cho GMM để phục vụ cho quá trình đào tạo và lưu giá trị cụm tương ứng cho mỗi sản phẩm vào bộ dữ liệu.
4. **for** `item in dataset` **do**
   - `Lấy ra số cụm của sản phẩm`
   - `Tìm tất cả những sản phẩm có cùng số cụm với sản phẩm truy vấn`
   - `Áp dụng công thức` (13) `để tính` $S(i,j)$ `cho mỗi cặp sản phẩm`
   - `Trả về top` m `sản phẩm tương tự bằng cách sắp xếp theo thứ tự giảm dần`

end

IV. KẾT QUẢ THỰC NGHIỆM

Ở mục này, chúng tôi sẽ chứng minh tính đúng đắn và hiệu quả mô hình đề xuất. Bằng cách so sánh mô hình đề xuất của chúng tôi với hai thuật

toán rất phổ biến và hiệu quả khác trong các hệ thống CB là BOW+GFF, GMM+ED. Chúng tôi cũng chứng minh ảnh hưởng của GMM lên độ chính xác và tính hiệu quả của GFF trong việc tính toán mức độ tương đồng so với cosine và euclidean.

A. Phương pháp đánh giá

Phương pháp đánh giá phổ biến của các hệ thống khuyến nghị thường là Mean Square Error (MSE, trung bình bình phương lỗi) là giá trị trung bình của tổng bình phương lỗi [27]. MSE càng nhỏ kết quả dự đoán đầu ra càng gần với kết quả thực tế. Nó được định nghĩa như sau:

$$MSE = \frac{1}{N}\sum_{1}^{n}(r_i - \widehat{r}_i)^2, \qquad (14)$$

trong đó, r_i vector biểu diễn sản phẩm được dự đoán ra, và $\widehat{r}_i$ là vector gốc biểu diễn sản phẩm truy vấn.

Chúng tôi cũng sử dụng kết quả khuyến nghị từ Sakenowa như một thước đo chuẩn để so sánh với 3 thuật toán thực nghiệm bởi vì Sakenowa là website rất có uy tín, tính phổ biến, nổi tiếng tại Nhật Bản trong rất nhiều năm. Bên cạnh đó, kết quả khuyến nghị của Sakenowa cũng vô cùng ấn tượng về độ chính xác và là một dịch vụ được tin cậy lâu dài trong thực tế. Người đọc có thể truy vấn tại Sakenowa tại đây https://sakenowa.com/

B. Phân tích thực nghiệm

Trong phần này, một vài thí nghiệm sẽ được tiến hành để kiểm chứng ảnh hưởng của GMM trong hệ thống gợi ý theo phân phối thuộc tính. Mô hình đề xuất của chúng tôi được thực hiện qua những bước như thống kê dữ liệu, làm sạch dữ liệu, nhóm tất cả sản phẩm vào các cụm khác nhau bằng GMM và cuối cùng sử dụng GFF và LD để sắp xếp và trả về kết quả.

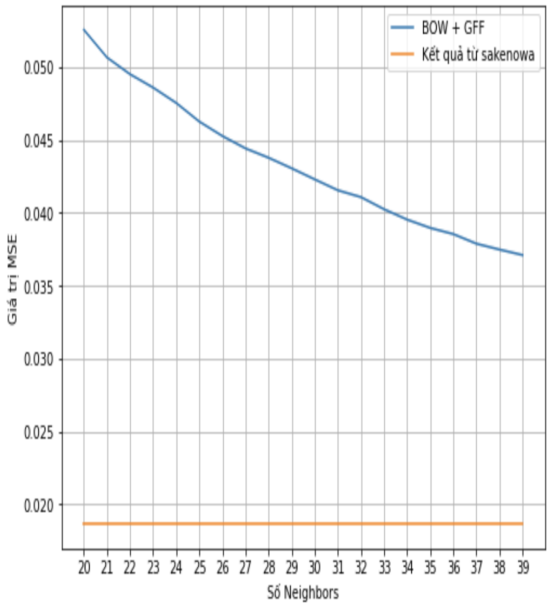
Để chứng thực sự hiệu quả của GMM và GFF cho kết quả dự đoán tốt hơn, chúng tôi chia thí nghiệm thành 3 phần. Đầu tiên, chúng tôi không sử dụng GMM, thay vào đó là thuật toán Bag-of-word (BOW) [1] trên một số thuộc tính như tags mùi vị trước khi áp dụng GFF để sắp xếp kết quả. Ở thí nghiệm thứ 2, chúng tôi áp dụng GMM+ED để làm rõ tác dụng của GMM. Và cuối cùng chúng tôi thí nghiệm mô hình đề xuất của mình để chứng minh hiệu quả của GMM và GFF, đồng thời đưa ra so sánh và đánh giá cho kết quả thí nghiệm.

1) Thí nghiệm 1: BOW + GFF: Lý do cho thí nghiệm này là để xác thực tác động của GMM lên độ chính xác của kết quả đầu ra so với thuật toán BOW. Do đó, ở thí nghiệm này chúng tôi áp dụng BOW kết hợp với GFF để cho kết quả đầu ra. Đầu tiên, chúng tôi thực hiện tiền xử lý dữ liệu cho các dữ liệu văn bản như stemming, replace synonyms, filling missing data,... [22]. Như đã đề cập ở mục trước, các trường văn bản quan trọng được viết dưới ngôn ngữ Nhật, nên chúng tôi sử dụng một số công cụ thư viện xử lý tiếng Nhật như Ginza [9], Janome [10], JapaneseStemmer [18], được lấy cảm hứng từ thuật toán Porter Stemming [28], để tiền xử lý.

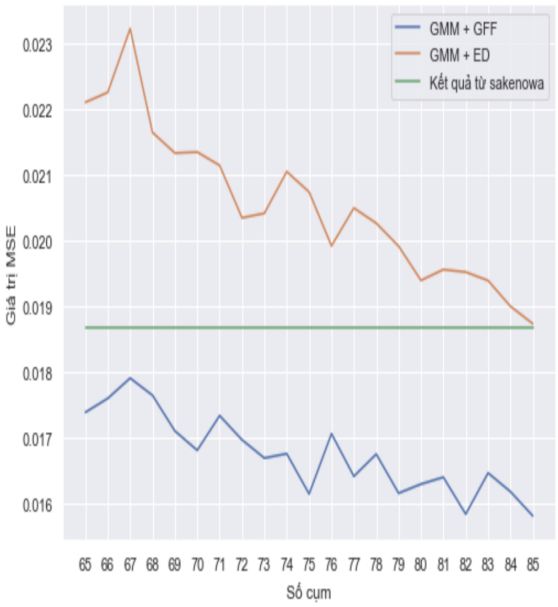
Trước khi sử dụng GFF cho việc sắp xếp kết quả, chúng tôi sử dụng BOW cho các trường văn bản đã được tiền xử lý để tìm ra ma trận vector biểu diễn cho các sản phẩm. Bước kế tiếp, chúng tôi sử dụng ma trận này như dữ liệu đầu vào cho thuật tóan K-nearest neighbors (KNN) dựa trên ý tưởng thuật toán không giám sát KNN Scikit-Learn [20] để tìm ra top các sản phẩm tương đồng nhau dựa vào các vectors này. Trong top các sản phẩm này, chúng tôi áp dụng công thức $S(i, j)$ trong (13) để lấy ra những kết quả tốt nhất.

2) Thí nghiệm 2: GMM + ED: Ở mục này, chúng tôi sẽ tận dụng GMM để gom nhóm n sản phẩm vào k nhóm. Tuy nhiên, đầu tiên chúng tôi vẫn áp dụng các bước tiền xử lý cho dữ liệu văn bản như ở Thí nghiệm 1. Sau đó, chúng tôi xây dựng một ma trận 6 chiều cho n sản phẩm, ma trận này biểu diễn cho các chỉ số 6 mùi vị và được đưa vào GMM để huấn luyện. Sau khi huấn luyện, kết quả cụm cho từng sản phẩm sẽ được lưu lại.

Ở bước tiếp theo, chúng tôi sẽ chuyển dữ liệu văn bản tags mùi vị thành ma trận biểu diễn các từ dưới dạng tần suất xuất hiện của từng từ trong toàn bộ danh sách tags mùi vị bằng cách sử dụng CountVectorizer của Scikit-Learn [20], và ghép với ma trận $(n, 6)$ bên trên để có được vector cuối cùng biểu diễn đặc trưng cho từng sản phẩm. Bước cuối cùng, để trả về được top sản phẩm tương tự nhất với 1 sản phẩm đầu vào, chúng tôi chỉ cần tìm đến cụm chứa sản phẩm đó và áp dụng công thức ED rồi sắp xếp kết quả trả về.



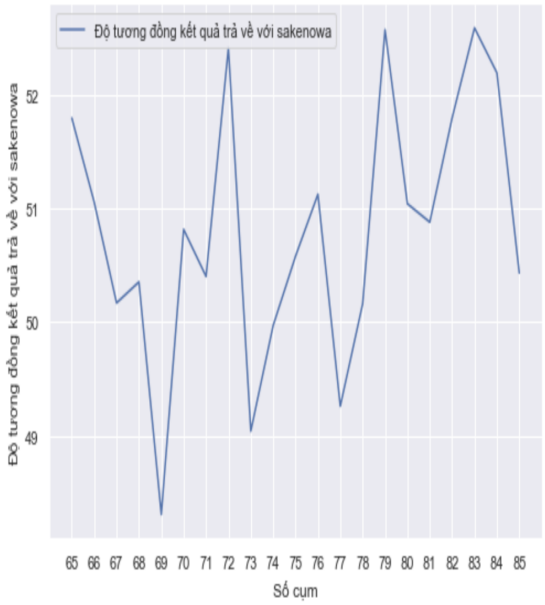
Hình 7: MSE áp dụng BOW+GFF



Hình 8: MSE áp dụng GMM+ED, GMM+GFF

3) Thí nghiệm 3: GMM + GFF: Hai thí nghiệm trên của chúng tôi là để chứng minh tầm quan trọng của GMM và GFF trong mô hình đề xuất của chúng tôi ở thí nghiệm này. Tương tự như xử lý trên, chúng tôi vẫn thực hiện các bước tiền xử lý dữ liệu văn bản như ở hai thí nghiệm trước. Tiếp theo chúng tôi xây dựng một ma trận $(n, 6)$ biểu diễn 6 chỉ số mùi vị cho n sản phẩm và đưa vào GMM như dữ liệu đầu vào để đào tạo. Lưu lại các giá trị cụm tương ứng của từng sản phẩm.

Để gợi ý ra những sản phẩm tương đồng nhất với một sản phẩm, chúng tôi chỉ cần tìm đến cụm mà sản phẩm này thuộc về và coi nó như trung tâm của cụm đó rồi sử dụng công thức (13) từng cặp từng cặp với các sản phẩm khác trong cụm đó. Sắp xếp các giá trị thu được theo thứ tự giảm dần chúng ta sẽ thu được kết quả tốt nhất từ hệ thống khuyến nghị.



Hình 9: Biểu đồ thống kê mức độ tương đồng kết quả với sakenowa

C. Kết quả thí nghiệm và so sánh

Tại phần này, chúng tôi so sánh thuật toán đề xuất của mình với kết quả khuyến nghị từ Sakenowa và 2 thuật toán CB khác. Chúng tôi kết luận rằng độ chính xác thuật toán của chúng tôi là tốt hơn Sakenowa và hai thuật toán còn lại. Kết quả so sánh được thể hiện trong Hình 7, Hình 8 và Hình 9.

Cả 3 thí nghiệm của chúng tôi đều trả về top 10 sản phẩm gần nhất cho mỗi sản phẩm trong bộ dữ liệu. Kết quả khuyến nghị từ Sakenowa cho mỗi sản phẩm được trả về từ bộ API[2]; trong đó, $f_{1...6}$ là giá trị tương ứng cho từng chỉ số mùi vị.

Ở Hình 7, danh sách giá trị của MSE được hiện thị và chịu ảnh hưởng bởi các số neighboors khác

[2]https://sakenowa.com/api/v1/brands/flavor?f=0&$fv = f_1, f_2, f_3, f_4, f_5, f_6$

nhau trong khoảng [25-39] của KNN. Chúng ta có thể dễ dàng nhận ra, mặc dù có xu hướng giảm, nhưng nó là không đáng kể và thời gian cho một lần tính toán là rất chậm do số neighboors tăng lên.

Tại Hình 8, khoảng cách của MSE giữa GMM+ED và GMM+DFF được hiển thị. Dựa vào biểu đồ này, có thể thấy GMM+GFF cho kết quả tốt hơn so với GMM+ED và chứng minh được tác dụng của GFF trong việc so sánh mức độ tương đồng. Cả 2 thí nghiệm đều hiển thị sự ảnh hưởng của số cụm GMM lên MSE trong khoảng [65-85].

Tại Hình 9, biểu đồ so sánh kết quả dự đoán trên toàn bộ dữ liệu của mô hình chúng tôi với kết quả gợi ý từ Sakenowa và xây dựng một danh sách thống kê phần trăm tương đồng qua từng giá trị cụm khác nhau.

Ở Bảng II và Bảng III, chúng tôi xây dựng bảng thống kê giá trị của MSE được sinh ra từ GMM+ED, BOW+GFF, GMM+GFF và kết quả từ Sakenowa. Dựa vào bảng thống kê này có thể nhận thấy rằng thuật toán GMM+GFF của chúng tôi cho kết quả tốt hơn hoàn toàn so với các thuật toán còn lại, và chứng minh được tính hiệu quả của thuật toán đề xuất trên bộ dữ liệu. Thêm nữa, thời gian xử lý của chúng tôi được thể hiện trong Bảng IV cũng cho thấy tốt hơn và nhanh hơn so với các thuật toán được đề xuất trước đây.

Bảng II: Giá trị MSE theo số lượng cụm GMM khác nhau

Số cụm	**GMM+ED**	**GMM+GFF**	**Sakenowa**
65	0.02211	**0.01738**	0.01868
70	0.02135	**0.01680**	0.01868
75	0.02074	**0.01613**	0.01868
80	0.01939	**0.01628**	0.01868
85	0.01873	**0.01580**	0.01868

Bảng III: Giá trị MSE ảnh hưởng bởi số neighboors trong KNN

Số neighbors	**BOW+GFF**	**Sakenowa results**
20	0.05254	0.01868
25	0.04624	0.01868
30	0.04228	0.01868
35	0.03895	0.01868
39	0.03709	0.01868

Bảng IV: Thời gian dự đoán cho một lần thực hiện

	BOW+GFF	**GMM+ED**	**GMM+GFF**
Thời gian	0.1856s	0.0174s	**0.0156s**

V. Kết luận

Ở bài báo này, chúng tôi đã đề xuất một thuật toán hiệu quả cho các bài toán gợi ý dựa theo phân phối thuộc tính trong các hệ thống khuyến nghị sử dụng thuật toán CB, và ứng dụng cho việc giải quyết bài toán gợi ý rượu trong một ứng dụng thực đang triển khai ở Nhật bản. Ngoài ra, chúng tôi cũng đề xuất một công thức sắp xếp mới cho danh sách kết quả tiềm năng thay vì sử dụng các công thức phổ biến như Cosine hay Euclidean.

Thuật toán đề xuất không chỉ đạt được độ chính xác cao, mà còn đạt được tốc độ xử lý rất nhanh phù hợp với các ứng dụng thực tế. Thuật toán hoàn toàn có thể áp dụng cho nhiều hoặc ít hơn 6 thuộc tính ở các bộ dữ liệu khác thay vì như thí nghiệm trên bộ dữ liệu về rượu của chúng tôi. Mặc dù có rất nhiều ưu điểm, tuy nhiên điểm hạn chế của thuật toán là cần huấn luyện lại mô hình sau khi có thêm một lượng các sản phẩm mới được thêm vào. Hướng nghiên cúu trong tương lai của chúng tôi là tìm cách cải thiện mô hình GMM trong khâu phân cụm sản phẩm để đạt được kết quả tốt hơn nữa.

Tài liệu tham khảo

[1] Sounak Bhattacharya and Ankit Lundia. “Movie Recommendation System Using Bag Of Words and Scikit-learn”. In: *International Journal of Engineering Applied Sciences and Technology* 04 (Oct. 2019), pp. 526–528. DOI: 10.33564/IJEAST.2019.v04i05.076.

[2] Dirk Bollen, Bart Knijnenburg, and Mark Willemsen. “Understanding choice overload in recommender systems”. In: Jan. 2010, pp. 63–70. DOI: 10.1145/1864708.1864724.

[3] Rui Chen, Qingyi Hua, and Gao. “A Hybrid Recommender System for Gaussian Mixture Model and Enhanced Social Matrix Factorization Technology Based on Multiple Interests”. In: *Mathematical Problems*

in Engineering 2018 (Oct. 2018), pp. 1–22. DOI: 10.1155/2018/9109647.

[4] Kong Fan-sheng. "Hybrid Gaussian pLSA model and item based collaborative filtering recommendation". In: *Computer Engineering and Applications* (2010).

[5] Dilan Görür and Carl Rasmussen. "Dirichlet Process Gaussian Mixture Models: Choice of the Base Distribution". In: *J. Comput. Sci. Technol.* 25 (July 2010), pp. 653–664. DOI: 10.1007/s11390-010-9355-8.

[6] Vairaprakash Gurusamy and Subbu Kannan. "Preprocessing Techniques for Text Mining". In: Oct. 2014.

[7] Ido Guy and David Carmel. "Social Recommender Systems". In: Jan. 2011, pp. 283–284. DOI: 10.1145/1963192.1963312.

[8] Rishin Haldar and Debajyoti Mukhopadhyay. "Levenshtein Distance Technique in Dictionary Lookup Methods: An Improved Approach". In: *Computing Research Repository - CORR* (Jan. 2011).

[9] Mai Hiroshi and Masayuki. "Ginza NLP Library". In: *25* (2019). URL: http://www.anlp.jp/proceedings/annual_meeting/2019/pdf_dir/F2-3.pdf.

[10] Janome$_{py}$. *Janome*. 2019. URL: https://github.com/mocobeta/janome.

[11] Shah Khusro, Zafar Ali, and Irfan Ullah. "Recommender Systems: Issues, Challenges, and Research Opportunities". In: Feb. 2016, pp. 1179–1189. ISBN: 978-981-10-0556-5. DOI: 10.1007/978-981-10-0557-2_112.

[12] Dar-Shyang Lee, Jonathan Hull, and B. Erol. "A Bayesian framework for Gaussian mixture background modeling". In: vol. 3. Oct. 2003, pp. III–973. DOI: 10.1109/ICIP.2003.1247409.

[13] Leo Liberti, Carlile Lavor, and Maculan. "Euclidean Distance Geometry and Applications". In: *SIAM Review* 56 (May 2012). DOI: 10.1137/120875909.

[14] Pasquale Lops, Marco de Gemmis, and Giovanni Semeraro. "Content-based Recommender Systems: State of the Art and Trends". In: Jan. 2011, pp. 73–105. DOI: 10.1007/978-0-387-85820-3_3.

[15] Linyuan Lü, Matúš Medo, and Chi Ho Yeung. "Recommender systems". English. In: *Physics Reports* 519.1 (Oct. 2012), pp. 1–49. ISSN: 0370-1573. DOI: 10.1016/j.physrep.2012.02.006.

[16] Yang Lu, Xuemei Bai, and Feng Wang. "Music Recommendation System Design Based on Gaussian Mixture Model". In: *ICM 2015*. 2015.

[17] Prem Melville and Vikas Sindhwani. "Recommender Systems". In: Jan. 2011, pp. 829–838. DOI: 10.1007/978-0-387-30164-8_705.

[18] MrBrickPanda. *Japanese Stemmer*. 2019. URL: https://github.com/MrBrickPanda/Japanese-stemmer.

[19] Suphakit Niwattanakul, Jatsada Singthongchai, and Naenudorn. "Using of Jaccard Coefficient for Keywords Similarity". In: Mar. 2013.

[20] Fabian Pedregosa, Alexandre Varoquaux, and Michel. "Scikit-learn: Machine learning in Python". In: *Journal of machine learning research* 12.Oct (2011), pp. 2825–2830.

[21] Simon Philip, Peter Shola, and Ovye Abari. "Application of Content-Based Approach in Research Paper Recommendation System for a Digital Library". In: *International Journal of Advanced Computer Science and Applications* 5 (Oct. 2014). DOI: 10.14569/IJACSA.2014.051006.

[22] Reza Rahutomo, Febrian Lubis, and Muljo. "Preprocessing Methods and Tools in Modelling Japanese for Text Classification". In: Aug. 2019. DOI: 10.1109/ICIMTech.2019.8843796.

[23] Martin Rajman and Romaric Besançon. "Text Mining: Natural Language techniques and Text Mining applications". In: *Proceedings of the 7th IFIP Working Conference on Database Semantics (DS-7)* (Jan. 1997). DOI: 10.1007/978-0-387-35300-5_3.

[24] Carl Rasmussen. "The Infinite Gaussian Mixture Model". In: vol. 12. Apr. 2000, pp. 554–560.

[25] Douglas Reynolds. "Gaussian Mixture Models". In: *Encyclopedia of Biometrics*

(Jan. 2008). DOI: 10.1007/978-0-387-73003-5_196.

[26] Philip Sedgwick. "Pearson's correlation coefficient". In: *BMJ* 345 (July 2012), e4483–e4483. DOI: 10.1136/bmj.e4483.

[27] Guy Shani and Asela Gunawardana. "Evaluating Recommendation Systems". In: vol. 12. Jan. 2011, pp. 257–297. DOI: 10.1007/978-0-387-85820-3_8.

[28] Karen Sparck Jones and Peter Willett, eds. *Readings in Information Retrieval*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1997. ISBN: 1558604545.

[29] Hangyu Yan and Yan Tang. "Collaborative Filtering based on Gaussian Mixture Model and Improved Jaccard Similarity". In: *IEEE Access* PP (Aug. 2019), pp. 1–1. DOI: 10.1109/ACCESS.2019.2936630.

[30] Kazuyoshi Yoshii, Masataka Goto, and Kazunori Komatani. "Hybrid Collaborative and Content-based Music Recommendation Using Probabilistic Model with Latent User Preferences." In: Jan. 2006, pp. 296–301.

[31] Bo Zhu, Jesus Bobadilla, and Fernando Ortega. "Reliability quality measures for recommender systems". In: *Information Sciences* (May 2018).

[32] B. Ziolko, Jakub Gałka, and Dawid Skurzok. "Modified Weighted Levenshtein Distance in Automatic Speech Recognition". In: Jan. 2010.

[33] Harry Zisopoulos, Savvas Karagiannidis, and Demirtsoglou. "Content-Based Recommendation Systems". In: (Nov. 2008).

A PROPOSAL OF ROBUST CONTENT-BASED RECOMMENDATION SYSTEM USING GAUSSIAN MIXTURE MODEL

***Tóm tắt*—Recommendation systems play an very important role in boosting purchasing consumption for many manufacturers by helping consumers find the most appropriate items. Furthermore, there is quite a range of recommendation algorithms that can be efficient; however, a content-based algorithm is always the most popular, powerful, and productive method taken at the begin time of any project. In the negative aspect, somehow content-based algorithm results accuracy is still a concern that correlates to probabilistic similarity. In addition, the similarity calculation method is another crucial that affect the accuracy of content-based recommendation in probabilistic problems. Face with these problems, we propose a new content-based recommendation based on the Gaussian mixture model to improve the accuracy with more sensitive results for probabilistic recommendation problems. Our proposed method experimented in a liquor dataset including six main flavor taste, liquor main taste tags, and some other criteria. The method clusters n liquor records relied on n vectors of six dimensions into k group ($k < n$) before applying a formula to sort the results. Compared our proposed algorithm with two other popular models on the above dataset, the accuracy of the experimental results not only outweighs the comparison to those of two other models but also attain a very speedy response time in real-life applications.**

***Từ khóa*—Recommendation system, Content-Based, Gaussian Mixture Model - GMM, Gaussian Filter Function, Collaborative Filtering.**

Nguyễn Văn Đạt đang theo học Thạc sĩ Khoa học Máy tính tại Đại học công nghệ Đại học quốc gia hà nội, đã tốt nghiệp bằng Kỹ sư Phần mềm tại trường Đại học Lê Quý Đôn năm 2017.

Lĩnh vực nghiên cứu là thị giác máy và các hệ thống khuyến nghị.

Tạ Minh Thanh nhận bằng kỹ sư CNTT và Thạc sĩ Khoa học Máy tính của Học viện Phòng vệ Nhật Bản, vào năm 2005 và 2008. Ông Thanh là giảng viên của trường Đại học Lê Quý Đôn từ năm 2005. Năm 2015, ông nhận bằng Tiến sĩ Khoa học Máy tính của Học viện Công nghệ Tokyo, Nhật Bản. Ông đã được công nhận chức danh Phó giáo sư của Hội đồng Giáo sư nhà nước vào năm 2019. Ông cũng là thành viên của Hiệp hội IPSJ Nhật Bản và Hiệp hội IEEE.

Lĩnh vực nghiên cứu của ông thuộc lĩnh vực thủy vân số, công nghệ mạng, bảo mật thông tin và thị giác máy.