CÔNG THỨC BAYES VÀ ỨNG DỤNG ĐỂ GIẢI QUYẾT CÁC BÀI TOÁN NHẬN DẠNG

Từ Trung Hiếu – (*Đại học Thủy lợi)*

1. **Công thức Bayes**

Theo suy nghĩ thông thường, nếu ta tìm được một hình ảnh E giống với một ký hiệu H mà ta đã biết trước đó, ta sẽ kết luận E là hình ảnh của H. Nhưng khi ta nhận thấy rằng E có thể hao hao giống H_1 hoặc H_2, ta sẽ phải sử dụng thêm các thông tin khác. Ví dụ như tần suất xuất hiện của H_1 và H_2, nếu ký hiện nào có tần suất lớn hơn, ta sẽ chọn ký hiệu đó. Hoặc dựa vào các hình lân cận của E để quyết định xem chọn H_1 hay H_2 là phù hợp. Đó là tất cả những gì mà Bayes đã phát biểu trong công thức.

$$p(H \mid E) = \frac{p(E \mid H).p(H)}{p(E)}$$

Như vậy, khả năng giả thuyết H ứng với bằng chứng E, tức là lượng *p(H|E),* phụ thuộc vào độ khớp của E đối với H, hay là lượng *p(E|H),* và tần suất xuất hiện của H, tức là lượng *p(H),* và bản chất của E, hay chính là lượng *p(E)*. Để chọn ra giả thuyết tốt nhất đối với mỗi E, chúng ta sẽ chọn ra H* có *p(H*|E)* cao nhất, cũng có nghĩa là lượng *p(E|H).p(H)* lớn nhất, vì lượng *p(E)* là cố định với mỗi E.

$$H^* = \arg\max_{H_k} p(H_k \mid E) = \arg\max_{H_k} \frac{p(E \mid H_k).p(H_k)}{p(E)} = \arg\max_{H_k} p(E \mid H_k).p(H_k)$$

Ví dụ trong ứng dụng quay số bằng giọng nói, người dùng nói ra một đoạn âm thanh A và máy cần tính toán để tìm ra một tên người N* khớp nhất với đoạn âm thanh vừa nhận được. Với giả sử trong trong máy tính có lưu các tên người N_1, N_2, N_K trong danh bạ. Nó sẽ giả định rằng N_1 cũng có thể là A, N_2 cũng có thể là A, do đó nó phải tính tất cả các giả định hay tính tất cả các lượng sau

$$p(N_1 \mid A) = p(N_1 \mid A).p(N_1) = equal(N_1, A).freq(N_1)$$
$$p(N_2 \mid A) = p(N_2 \mid A).p(N_2) = equal(N_2, A).freq(N_2)$$
...
$$p(N_K \mid A) = p(N_K \mid A).p(N_K) = equal(N_K, A).freq(N_K)$$

Trong đó equal(N_k, A) là độ giống nhau giữa N_k và A. Khi N_k càng giống A thì độ đo này tiến dần về 1. Khi N_k càng khác A thì con số này tiến dần về 0. Sau đó nó sẽ chọn ra N_k nào có $p(N_k \mid A)$ là lớn nhất. Trong trường hợp các khả năng xuất hiện của các tên là như nhau, nghĩa là các $p(N_k)$ đều bằng nhau, thì khả năng N_k là A chính là độ khớp của N_k với A. Đây là trường hợp đặc biệt của công thức Bayes, trong đó thông tin về tần xuất của các giả thuyết không đóng góp gì vào nhận dạng.

2. Nhận dạng một ký hiệu đơn

Một ký hiệu (symbol) trong nhận dạng thường được dùng để chỉ một đơn vị độc lập có thể được đưa vào các phép so sánh để đem lại kết quả nhận dạng. Trong nhận dạng tiếng nói,

một ký hiệu thường ứng với một âm tiết (syllable). Trong nhận dạng chữ viết, một ký hiệu có thể là một chữ đơn (character), nếu ta chia được từ thành chữ, hoặc một từ (handwritten word) gồm nhiều chữ liền nét.

Trong nhận dạng một ký hiện đơn, ta cần một từ điển D các mẫu nhận dạng. Từ điển này sẽ được tạo trong quá trình huấn luyện. Ta giả định từ điển D liệt kê được, nghĩa là nó hỗ trợ toán tử size(D) cho kích thước của từ điển và item(k, D) cho phần tử mẫu thứ k trong từ điển D. Do đó thủ tục nhận dạng sẽ như sau

b1) Ban đầu đặt giá trị kmax = -1; pmax = 0;
b2) Với mỗi giá trị item(k, D) có trong từ điển, ta tính lượng pk
pk = equal(item(k, D), V) * freq(item(k, D));
b3) Đặt lại giá trị kmax và pmax nếu như pk lớn hơn pmax
b4) Trả về giá trị kmax tìm được

Thủ tục tìm kiếm này sẽ trả về -1 trong trường hợp từ điển rỗng, và trả về kmax nằm trong khoảng 0 đến size(D)-1 với kmax có khả năng lớn nhất. Nếu chúng ta đặt ngưỡng ε cho việc nhận dạng, thủ tục tìm kiếm cũng trả về -1 khi pmax nhỏ hơn ε

Trong phương pháp nhận dạng này, từ điển D có nhiều phần tử, và ta dùng biểu thức item(k, D) để lấy phần tử thứ k. Mỗi phần tử là một mẫu (model) và việc nhận dạng thực chất là so sánh đối tượng cần nhận dạng V với các mẫu trong từ điển. Về mặt lập trình, mẫu nhận dạng là bất kỳ cấu trúc dữ liệu nào cho phép thực hiện hai toán tử equal và freq như trên. Dưới đây chúng tôi sẽ giới thiệu một số các phần tử cơ bản có thể dùng làm mẫu.

Dạng đơn giản nhất của mẫu M = (μ, δ, ρ) trong đó μ là một véc tơ gọi là tâm của mẫu, δ là một số thực dương xác định bán kính của mẫu, và ρ xác định khả năng xuất hiện của mẫu. Do đó ta có thể định nghĩa hàm equal như sau

$$equal(V, M) = \exp\left(-\frac{(V-\mu)^2}{2\sigma^2}\right) \quad \text{và} \quad freq(M) = \rho$$

Việc huấn luyện mẫu này được thực hiện bằng cách tính ba tham số μ, δ, ρ từ tập dữ liệu huấn luyện tương ứng. Đây chỉ là các phép toán thống kê thông thường trong đó μ được tính bằng trung bình của các mẫu huấn luyện, δ được tính bằng khoảng cách lớn nhất giữa μ và các mẫu, và ρ là số lượng mẫu có tâm μ trên tất cả các mẫu.

Mô hình thống kê HMM cũng hay được dùng làm phần tử nhận dạng. Một mô hình HMM thường có ba tham số λ=(A, B, Π) được mô tả trong các tài liệu [3, 2, 4]. Ta có thể tính lượng $equal(V, \lambda) = p(V|\lambda)$ thông qua thuật toán ước lượng. Và ta có thể lưu thông tin thống kê $p(\lambda)$ như trường hợp trên. Việc huấn luyện được thực hiện thông qua thuật toán Baum-Welch

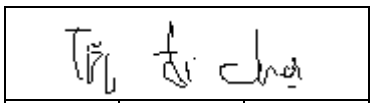
3. Nhận dạng các chuỗi ký hiệu rời rạc

Một chuỗi ký hiệu (symbol sequence) thường được dùng để chỉ một dãy tuần tự các ký hiệu được ghép nối liên tục với nhau, ví dụ như một chuỗi các âm tiết được phát ra, một dãy liên tục các từ được viết trên một dòng, một dãy các hình ảnh liền nhau trong một đoạn phim.

Chuỗi ký hiệu rời rạc (connected symbol sequence) là một chuỗi ký hiện trong đó các ký hiệu có các khoảng trống để có thể phân biệt được. Trong nhận dạng, khoảng trống cũng là một ký hiệu và thường là các vùng tín hiệu không mang năng lượng. Vì chuỗi tín hiệu rời rạc có thể chia nhỏ thành các ký hiệu độc lập (isolated symbols), bài toán nhận dạng chuỗi tín hiệu rời rạc được đưa về bài toán nhận dạng ký hiệu đơn. Tuy nhiên chúng ta hãy xem xét thuật toán nhận dạng với các bước sau

b1) Chia nhỏ chuỗi ký hiệu thành các ký hiệu tách biệt
b2) Áp dụng thuật toán nhận dạng ký hiệu riêng để tìm ra các ứng cử viên cho ký hiệu, mỗi một ký hiệu có một tập cac ứng cử viên có xác suất giả định cao nhất.
b3) Dùng thông tin ngữ cảnh hay thông tin ngôn ngữ để lựa chọn câu có khả năng xuất hiện cao nhất.

Chúng ta hãy xét một ví dụ đơn giản nhất để nhận dạng dãy các ký hiệu viết tay dưới đây để làm rõ thuật toán nhận dạng các ký hiệu rời rạc. Trong ví dụ dưới đây, chúng ta chia dòng chữ làm ba ký hiệu và nhận dạng được ba tập từ tương ứng. Việc lựa chọn câu nào từ ba tập từ phải sử dụng thông tin ngôn ngữ, hay cụ thể hơn là tần suất xuất hiện của một câu. Chúng ta sẽ thấy khả năng "Tôi đi chơi" hoặc "Tôi đi chợ" là rất cao, nhưng chúng ta sẽ không thấy "Tôi đi chợt" hoặc "Tra đi chợ" vì các câu nói đó xuất hiện rất ít hoặc không xuất hiện trong tiếng Việt.



Tôi	đi	chơi
Thôi	ti	chợ
Ta	si	chạ
Tra		chợt

Thông tin ngôn ngữ (language information) thường được lưu ở hai dạng phổ biến, mô hình ngôn ngữ (language model) và văn phạm (grammar) cùng với các hình thức tương đương văn phạm. Mô hình ngôn ngữ [2, 5, 6] là một công cụ thống kê cho phép tính xác suất của một câu nói trong ngôn ngữ. Các câu nói thường gặp sẽ có tần suất cao, các câu nói sai ngữ pháp hoặc ít gặp sẽ có xác suất xấp xỉ không. Mô hình ngôn ngữ phản ánh quy luật ngữ pháp, ngữ nghĩa, ngữ dụng dưới dạng thống kê. Văn phạm [7, 8, 11] và các dạng tương đương của nó phản ánh ngữ pháp của ngôn ngữ. Văn phạm là các quy tắc ghép ký hiệu chính xác và không thể sinh tự động như các quy luật thống kê, do đó chúng ta cần phải biên soạn các bộ văn phạm để phản ánh thông tin ngôn ngữ.

Mô hình ngôn ngữ thường được lưu thành mô hình bigram, trong đó mỗi từ có xác suất đứng đầu p(W) và xác xuất đứng sau một từ nào đó $p(W_{sau} \mid W_{truoc})$ do đó câu nói trên được xác định như sau, với giả định ta có ba ký hiệu Ttri, dsi, chci ứng với ba hình ảnh chưa biết. Ta sẽ tính các lượng như dưới đây và chọn ra câu có khả năng cao nhất. Ví dụ ta sẽ tính các lượng sau

equal(Tôi, Ttri) . equal(đi, dsi) . equal(chơi, chci) . p(Tôi) . p(đi | Tôi) . p(chơi | đi)
equal (Ta, Ttri) . equal (ti, dsi) . equal(chợ, chci) . p(Ta) . p(ti | Ta) . p(chợ | ti)

Trong đó các hàm equal được dùng để xác định độ khớp giữa các hình ảnh và mô hình của các từ. Các hàm xác suất phía sau được lấy từ mô hình ngôn ngữ. Chúng ta có thể thấy đây là công thức Bayes trên câu, p(câu | hình) = p(hình | câu) . p(câu) nhưng một câu được chia thành nhiều từ và một hình được chia thành nhiều ký hiệu đơn lẻ.

4. Nhận dạng các chuỗi ký hiệu liên tục

Chuỗi ký hiệu liên tục (continuous symbol sequence) là chuỗi ký hiệu trong đó ta không đủ thông tin để tách biệt các ký hiệu thành các từ đơn. Có nghĩa là các khoảng trống giữa các ký hiệu không tồn tại hoặc không đủ lớn để nhận ra, và do đó chúng ta không thể chia nhỏ các từ. Ví dụ các từ được nói liên tục trong bản tin thời sự hoặc bình luận bóng đá, hoặc ví dụ các từ được viết dày và liên tục trên một dòng và không thể chia nhỏ thành các từ đơn.

Khi chuỗi ký hiệu không thể chia nhỏ được, ta phải xử lý toàn bộ chuỗi ký hiệu và coi nó như một đối tượng hay một ký hiệu đơn. Có hai cách tiếp cận phổ biến cho việc nhận dạng chuỗi ký hiệu liên tục. Cách thứ nhất là tìm kiếm chuỗi cần nhận dạng trong không gian chuỗi mẫu. Có thể hiểu là tìm kiếm trên từ điển giống như nhận dạng ký hiệu đơn lẻ. Nhưng cũng có thể sử dụng thuật toán tìm chuỗi tối ưu, ví dụ thuật toán Viterbi [2, 3] để tìm chuỗi trạng thái khớp nhất với chuỗi cần nhận dạng. Cách thứ hai là dùng phương pháp tổng hợp từ dưới lên với các bộ phân tích cú pháp từ dưới lên được trình bày trong [9, 10, 11, 12] để sinh ra một cấu trúc cây trong đó có các từ thay thế và từ của ngôn ngữ. Cách này đòi hỏi phải biên soạn bộ văn phạm để các bộ phân tích có thể hoạt động.

Kết luận

Công thức Bayes là cơ sở để xác định khả năng của một giả định dựa trên bằng chứng. Khi có một đoạn dữ liệu S cần nhận dạng, ta cần giả định rằng S có thể khớp với bất kỳ một mẫu dữ liệu M_1, M_2, M_K nào đã biết trước đó. Do đó ta cần chọn một giả định tốt nhất bằng cách ước lượng khả năng hay xác suất của giả định đó bằng công thức Bayes. Công thức Bayes cũng được phát triển để nhận dạng các chuỗi ký hiệu. Trong đó xác suất tiên nghiệm, hay khả năng xuất hiện của một từ hoặc một câu, được xác định bằng thông tin ngôn ngữ, hay cụ thể hơn là mô hình ngôn ngữ.

Văn phạm là một giải pháp thay thế cho thông tin ngôn ngữ. Mặc dù các luật của văn phạm rất chặt chẽ, nhưng chúng ta cần biên soạn. Các luật thống kê trong mô hình ngôn ngữ có thể tạo một cách tự động, hơn nữa nó phản ánh cả ngữ pháp, ngữ nghĩa, và ngữ dụng của câu nói trong ngôn ngữ.

Tóm tắt

Các nghiên cứu về nhận dạng sử dụng phương pháp thống kê ngẫu nhiên thường sử dụng công thức Bayes để tính các xác suất của các giả định và lựa chọn giả định có xác suất cao nhất làm kết quả nhận dạng. Trong bài báo này, chúng tôi muốn giới thiệu một số dạng khác nhau của công thức Bayes và ứng dụng của nó trong các bài toán nhận dạng khác nhau. Qua đó chúng tôi cũng giới thiệu một số khái niệm như không gian mẫu, mô hình ngôn ngữ, văn phạm, mô hình Markov ẩn.

Từ khóa: Bayesian rule, speech recognition, handwriting recognition, language model, hidden markov model, context-free grammar.

Summary

Bayesian rule and its application to solve recognition problems

Tu Trung Hieu - { tutrunghieu@gmail.com }

Researches on recognition with stochastic approach usually use the Bayesian rule to evaluate the probabilities of hypotheses and select the hypothesis with the maximum probability to be the recognition result. In this paper, we would like to introduce the Bayesian rule and its application in different recognition problems. In addition, we also introduce some recognition concepts, such as pattern space, language model, grammar, hidden Markov model.

Tài liệu tham khảo

[1] E. T. Jaynes (2003), *Probability Theory: The Logic of Science*, Cambridge University Press.

[2] Steve Young, Dan Kershaw, Julian Odell, Dave Ollason, Valtcho Valtchev, Phil Woodland (2000), *The HTK Book.*

[3] Lawrence R. Rabiner, *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition.* Proceedings of the IEEE, 77 (2), p. 257–286, February 1989.

[4] Gernot A. Fink and Thomas Plötz (2007), *Markov Models for Handwriting Recognition*, ICDAR 2007 Tutorial, Curitiba, Brazil

[5] Fei Song, W. Bruce Croft (1999), *A General Language Model for Information Retrieval*.

[6] Jay M. Ponte, W. Bruce Croft (1998), *A Language Modeling Approach to Information Retrieval,*

[7] Jean-Michel Autebert, Jean Berstel, Luc Boasson ((1997), *Context-Free Languages and Push-Down Automata.*

[8] J.E. Hopcroft and J.D. Ullman (1979). *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley,

[9] Philippe Mclean. Nigel Horspool (1996), *A Faster Earley Parser.*

[10] Mark Hepple (1999), *An Earley-style Predictive Chart Parsing Method for Lambek Grammars.*

[11] Alon Lavie, Masaru Tomita (1993), *GLR* An Efficient Noise-skipping Parsing Algorithm For Context Free Grammars.*

[12] J. C. Chappelier, M. Rajman (1998), *A generalized CYK algorithm for parsing stochastic CFG.*