# Báo cáo Bài tập thực hành Xử lý đa chiều 04

Nguyễn Phú Thành - MSSV: 18110014

3/7/2021

**Đề bài:**

a) Read some article about the t-SNE. Source:

- Why You Are Using t-SNE Wrong
- How to Use t-SNE Effectively

b) and write a summary of the t-SNE (1-page summary)

**Bài làm:**

t-SNE (Stochastic Neighbor Embedding with t-distribution) là một thuật toán giúp giảm số chiều của dữ liệu, được dùng chủ yếu để trực quan hóa dữ liệu.

- SNE là thuật toán gốc trong đó thuật toán cố gắng tìm một phân phối q xấp xỉ gần với phân phối p bằng hàm loss Kullback-Leibler:

$$C = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

trong đó:

$$\begin{aligned} p_{j|i} &= \frac{\exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2/2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_k\|^2/2\sigma_i^2)} \\ p_{ij} &= \frac{p_{i|j} + p_{j|i}}{2N}, q_{ij} = \frac{\exp(-\|\boldsymbol{y}_i - \boldsymbol{y}_j\|^2)}{\sum_k \sum_{l \neq k} \exp(-\|\boldsymbol{y}_k - \boldsymbol{y}_l\|^2)} \end{aligned}$$

- Thuật toán t-SNE là một biến thể của thuật toán SNE, thay phân phối Gauss trong q sang phân phối Student để giải quyết "crowding problem" trong SNE

$$q_{ij} = \frac{(1 + \|\boldsymbol{y}_i - \boldsymbol{y}_j\|^2)^{-1}}{\sum_k \sum_{l \neq k} (1 + \|\boldsymbol{y}_k - \boldsymbol{y}_l\|^2)^{-1}}$$

Có ba hyperparameter quan trọng trong thuật toán t-SNE là:

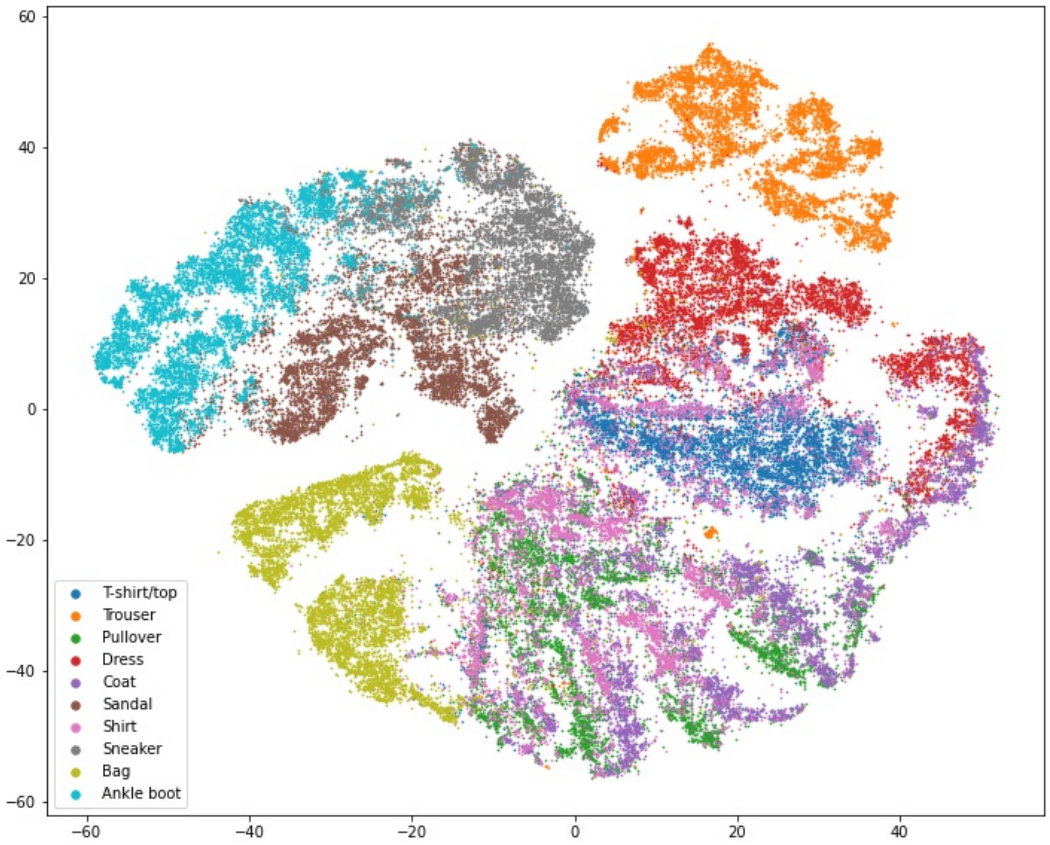
- Hyperparameter Perplexity: Cân bằng độ quan trọng giữa thông tin lân cận và thông tin toàn cục của dữ liệu, hay nói cách khác, ta có thể coi tham số này là để ước lượng số lân cận xung quanh mỗi điểm dữ liệu. Perplexity nhỏ thì kết quả t-SNE sẽ cho thấy cấu trúc lân cận của các điểm dữ liệu, perplexity lớn cho thấy rõ hơn về cấu trúc toàn cục của dữ liệu và làm cho cấu trúc lân cận mờ đi
  Nên thực hiện t-SNE nhiều lần với các perplexity khác nhau để xem các phân cụm thay đổi như thế nào để có thể đưa ra kết luận. Hyperparameter perplexity nên chọn trong khoảng từ 5-50 theo nhóm tác giả đưa ra thuật toán t-SNE, van der Maaten và Hinton

- Hyperparameter Learning rate: Tốc độ học của thuật toán ở mỗi lần cập nhật tham số (t-SNE sử dụng thuật toán Gradient Descent để cập nhật các tham số nên đây là hyperparameter quan trọng). Learning rate quá nhỏ dẫn đến hội tụ chậm, learning rate quá lớn dẫn đến tham số dao động nhiều, khó hội tụ
- Hyperparameter n_iter: Số lần lặp (Số epoch). Nên điều chỉnh hyperparameter này sao cho các tham số của t-SNE đã hội tụ một cách ổn định

Các điểm cần lưu ý khi rút ra kết luận về kết quả của t-SNE:

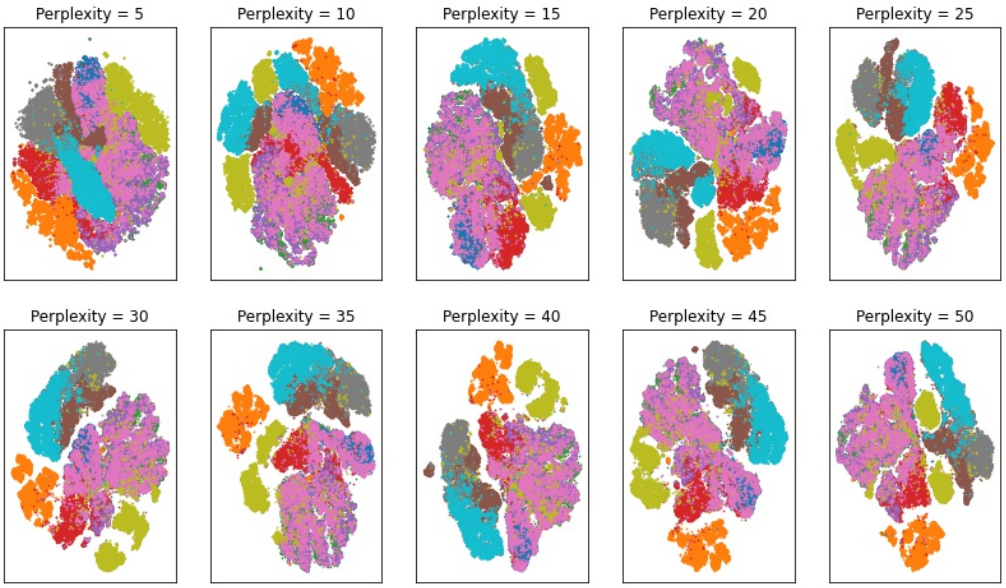
- Kích thước của các phân cụm trong t-SNE *không có ý nghĩa nhiều*
- Khoảng cách giữa các phân cụm với nhau trong t-SNE *không có ý nghĩa nhiều*
- Các cụm tách biệt trong t-SNE *có thể là do nhiễu*
- Để xem dạng topo của dữ liệu nhiều chiều, ta cần xem kết quả t-SNE với các perplexity khác nhau

c) Applying the t-SNE on Fashion-MNIST dataset
**Bài làm:** Do bộ dữ liệu Fashion-MNIST là khá lớn (60000 ảnh trong tập train và 10000 ảnh trong tập test, tức gộp lại ta có 70000 ảnh trong bộ dữ liệu) và thuật toán t-SNE chạy khá chậm theo cách cài đặt của sklearn, nên để giảm thời gian chạy (do ta phải thực hiện t-SNE nhiều lần với các giá trị perplexity khác nhau), ta sử dụng GPU do Google Colab cung cấp cùng với thư viện tsnecuda




Hình 1: t-SNE trên bộ dữ liệu Fashion MNIST

Hình trên là kết quả của t-SNE trên bộ dữ liệu Fashion MNIST với Perplexity = 40, n_iter = 5000 và learning_rate = 200
Trước khi đưa ra nhận xét, ta sẽ xem kết quả của t-SNE với các giá trị perplexity khác nhau (với cùng n_iter và learning_rate):




Hình 2: t-SNE trên bộ dữ liệu Fashion MNIST với perplexity khác nhau

Nhận thấy với perplexity đủ lớn ($\geq 25$), ta có thể đưa ra các kết luận sau:

- Bộ dữ liệu dễ dàng phân tách được các ảnh là Bag (Cụm màu vàng)
- Bộ dữ liệu dễ dàng phân tách được các ảnh là Trouser (Cụm màu cam)
- Các ảnh thuộc vào các nhóm: T-shirt, Shirt, Pullover, Coat và Dress thuộc vào một phân cụm riêng (Các cụm màu xanh biển, xanh lá, hồng, tím và đỏ)
- Các ảnh thuộc các nhóm: Sandal, Sneaker và Ankle Boot thuộc vào một phân cụm riêng (Các cụm màu xanh trời, nâu và xám)

**Tài liệu tham khảo:**

1. Các bước thực hiện cài đặt anaconda trên google colab (để cài đặt tsnecuda phía dưới) được tham khảo tại: https://towardsdatascience.com/conda-google-colab-75f7c867a522
2. Các bước thực hiện cài đặt tsnecuda: https://github.com/CannyLab/tsne-cuda/wiki/Installation
3. Các bài viết/slide bài giảng về t-SNE được tham khảo trong câu b:
   - Why You Are Using t-SNE Wrong
   - How to Use t-SNE Effectively
   - http://www.cs.toronto.edu/~hinton/absps/tsne.pdf
   - https://www.cs.toronto.edu/~jlucas/teaching/csc411/lectures/lec13_handout.pdf