**KẾT QUẢ CHẠY DEMO**

**TÁC VỤ DỊCH MÁY BẰNG TRANSFORMER TRÊN IWSLT-14**

- Em thực hiện training thử với chỉ 25 epochs để tiết kiệm thời gian thì kết quả thu được như sau:



+ Giá trị loss thu được trên tập train là 3.501 và giá trị negative log likelihood tương ứng của nó là 2.032.

+ Perlexity thu được trên tập train là 4.09



+ Trên tập validation sẽ có thêm giá trị bleu và bleu được hiển thị ở đây là BLEU-4.

+ Giá trị loss thu được là 3.879 và giá trị negative log likelihood tương ứng của nó là 2.335.

+ Perlexity là 5.05 và BLEU là 34.79%, còn rất cao do em chỉ chạy có 25 epochs để thử nghiệm.

- Một số kết quả từ các nguồn trên github khác thì quá trình training hoàn tất với ‘max peochs’ mặc định sẽ xấp xỉ 1 ngày khi training trên GPU-V100.

- Đánh giá bằng việc dịch trên tập test:

Ảnh có chứa văn bản, đen và trắng, Phông chữ, màu trắng

Mô tả được tạo tự động

+ Những mẫu số theo sau S, T, H, D, P chính là ID của câu trong tập dữ liệu, do quá trình load data và split data là random nên kèm theo ID khi xuất kết quả.

+ S là câu nguồn cần dịch.

+ T là câu đích được dịch do con người dịch bằng ngôn ngữ tự nhiên của con người.

+ H là câu được dịch bằng giả thuyết của mô hình, tức là có áp dụng các kĩ thuật khác như BPE, tokenizer trước khi được dịch bằng transformer hay không, nếu không có các đối số đó thì dùng mặc định của hàm.

+ D là câu được dịch khi giải mã ngược lại những gì được sử dụng từ H, nếu không truyền vào các đối số trên H thì D sẽ giống H.

+ P là giá trị xác suất của từng vị trí trong câu nhưng đã được lấy logarit cơ số 2, do xác suất sau khi đi qua softmax của mỗi từ được dịch sẽ có giá trị thuộc [0, 1] nên sẽ có giá trị âm.

+ Và những con số đi sau H và D chính là giá trị trung bình của P.

+ Với beam-size = 5 cho quá trình beam-search các từ phù hợp nhất trong mỗi lần dịch thì BLEU-4 trung bình cho mô hình đã được training chỉ với 25 epochs là 34.30%. Các giá trị theo sau là 67.9/42.3/28.2/19.2 theo em tìm hiểu thì có thể là BLEU-2/3/5/6. BLEU vẫn khá giống trên tập Validation.

- Em đang xem script python của các hàm trong fairseq!