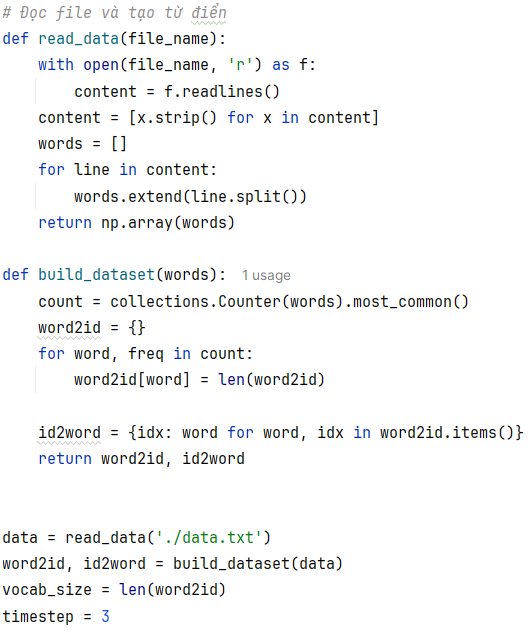
**BÀI TẬP 6**

**XÂY DỰNG MÔ HÌNH HỒI QUY (TH1)**

**--------**

1. **Xây dựng mô hình LSTM cho bài toán sinh từ:**
   1. **Đọc dữ liệu và tạo từ điển:**



*Hình. Đọc dữ liệu và tạo từ điển*

- Hàm **read\_data()** dùng để đọc nội dung từ file văn bản:

+ **content** sẽ lưu toàn nội dung file\_name và lưu từng dòng.

+ Thực hiện loại bỏ ký tự xuống dòng ở cuối mỗi dòng và gán lại vào biến **content**.

+ Tạo một danh sách **words** chứa các từ, và cũng là kết quả của hàm này.

+ Tách dòng thành các từ và thêm vào danh sách **words**.

+ Chuyển danh sách **words** thảnh mảng Numpy.

- Hàm **build\_dataset()** xây dựng tập từ điển từ danh sách chứa các từ:

+ Tạo danh sách từ và số lần xuất hiện, xếp theo thứ tự từ nhiều nhất đến ít nhất.

+ Tạo một từ điển **word2id** để ánh xạ từ sang chỉ số.

+ Tạo thêm một từ điển **id2word** để ánh xạ ngược từ chỉ số sang từ.

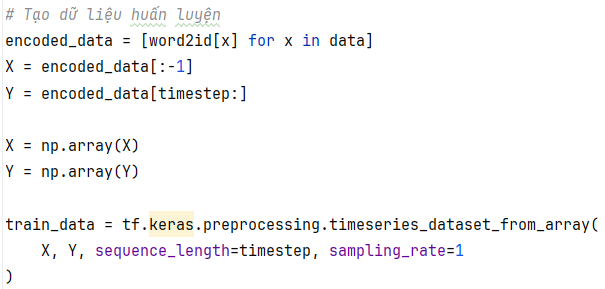
- Đọc nội dung file ‘data.txt’ và lưu vào biến **data**.

- Tạo từ điển **word2id** và **id2word** từ dữ liệu **data**.

- **vocab\_size** là số lượng từ trong từ điển **word2id**.

- **timestep** là số lượng từ trong một chuỗi đầu vào.

* 1. **Tạo dữ liệu huấn luyện:**



*Hình. Tạo dữ liệu huấn luyện*

- **encoded\_data** là danh sách các chỉ số đã được mã hóa toàn bộ dữ liệu **data**.

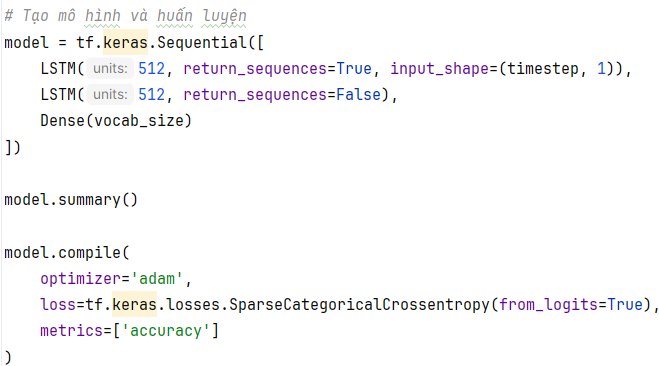
- **X**: Lấy tất cả từ đầu tiên đến trước từ cuối cùng.

- **y:** Lấy tất cả từ thứ timestep trở đi.

- Chuyển **X** và **y** thành mảng Numpy.

- **train\_data**: Tạo dataset chuỗi thời gian từ dữ liệu **X** và **y** để huấn luyện.

* 1. **Xây dựng mô hình:**



*Hình. Xây dựng mô hình*

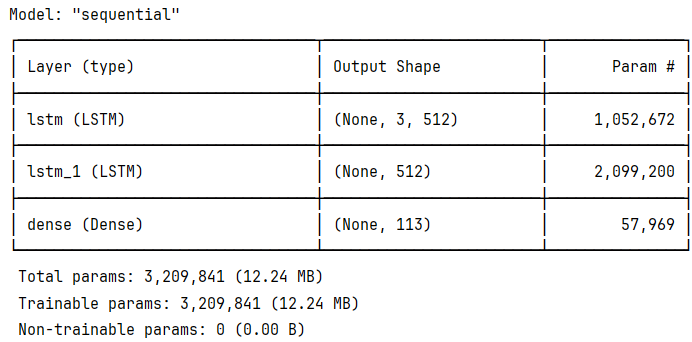
**-** Sử dụng lớp **Sequential()** để xây dựng mô hình:

+ Tầng **LSTM** đầu tiên có 512 units, tức là số tế bào nhớ. Tham số *return\_sequence* để trả về toàn bộ đầu ra theo từng timestep và *input\_shape* là timestep và số đặc trưng là ‘1’.

+ Tầng **LSTM** thứ hai cũng có 512 units, nhưng tầng này chỉ chuỗi đầu ra cuối cùng.

+ Tầng đầu ra **Dense** có số nơ-ron là **vocab\_size**. Tầng này sẽ lấy đầu ra từ tầng **LSTM** và chuyển thành giá trị dự báo mong muốn.

- Biên dịch mô hình với optimizer Adam, hàm lỗi SparseCategoricalCrossentropy và sử dụng độ chính xác làm thước đo hiệu suất.



*Hình. Kiến trúc mô hình*

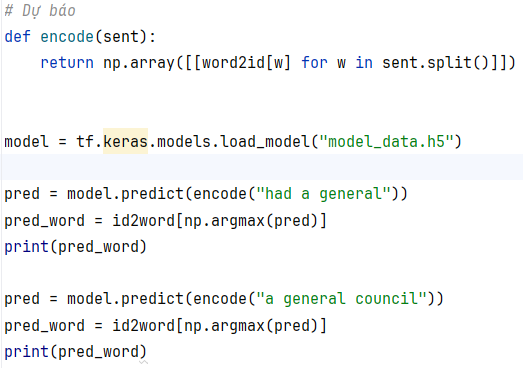
* 1. **Huấn luyện mô hình:**



*Hình. Huấn luyện mô hình*

Mô hình sẽ được huấn luyện qua 500 epochs với dữ liệu huấn luyện **train\_data**.

* 1. **Dự báo:**



*Hình. Dự báo*

- Hàm **encode()** được sử dụng để chuyển câu thành mảng các chỉ số. Hàm trả về mảng Numpy sau khi đã tách câu thành từ, mã hóa từng từ.

- Tải mô hình đã lưu ‘model\_data.h5’ và lưu vào **model**.

- **pred** lưu giá trị dự đoán tiếp theo sau câu “had a general’.

- **pred\_words** sẽ tìm chỉ số xác suất cao nhất và chuyển đổi thành từ dựa trên từ điển **id2word**.

- Trường hợp ‘a general council’ tương tự như trên.

**--- HẾT ---**