Documentație proiect NLP - "Image Captioning"

Nicolae Ducal, Alexandru Ioan Țifui, Daniel Avram

Introducere

Tema abordată în cadrul proiectului a fost "Image Captioning". Aceasta constă în generarea unei descrieri potrivită pentru conținutul unei imagini date ca input.

"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

Arhitectura folosită are la bază ideea din lucrarea "Show and Tell"[3]. Anume aceasta presupune extragerea feature-urilor dintr-o imagine folosind o rețea convoluțională (CNN) iar apoi trecerea acestora printr-o rețea neuronală recursivă (RNN) pentru obținerea descrierei.

CNN

Pentru partea de CNN am folosit modelul "Inception-V3"[1] preantrenat pe setul de date "ImageNet" oferit de PyTorch. Aceasta primește imagini de dimensiune 299×299 și returnează feature-uri de dimensiune $8 \times 8 \times 2048$ (trebuie notat că am ignorat ultimul strat

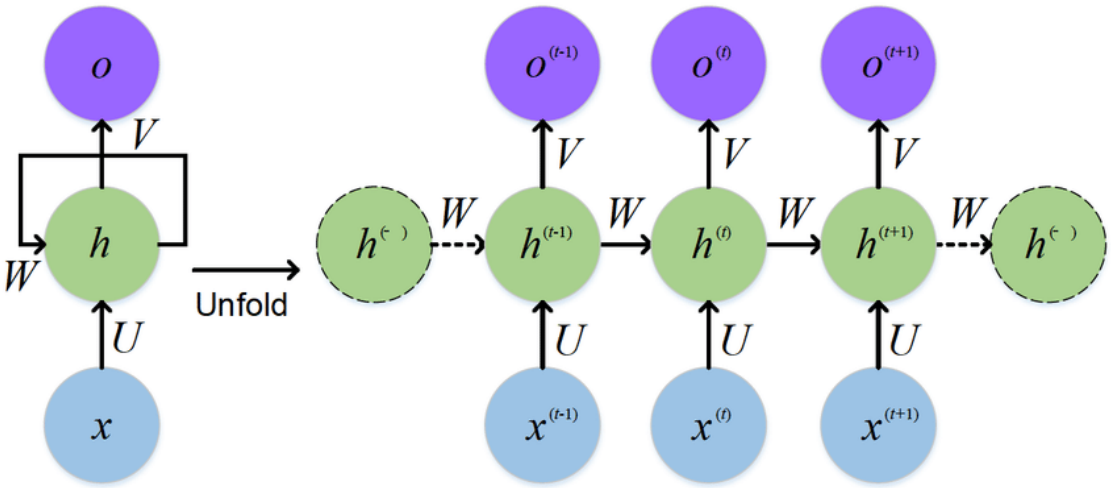
de tip "Fully-Connected" al rețelei). Pentru a facilita lucrul ulterior cu rețeaua RNN, am redimensionat feature-urile la 64×2048.

După extragerea feature-urilor am adăugat un strat liniar pentru a redimensiona feature-urile la $64 \times Embedding\_Size$, unde $Embedding\_Size$ reprezintă dimensiunea scufundării folosită de rețeaua RNN.

RNN

Rețelele RNN[2] sunt un tip rețele neuronale folosite pentru caracterizarea datelor secvențiale. Datele secvențiale pot fi privite ca niște vectori în care fiecare valoare depinde de cele precedente ei. În cazul problemelor care implică generare de text, fiecare cuvânt depinde de cele precedente atât din punct de vedere semantic cât și sintactic.

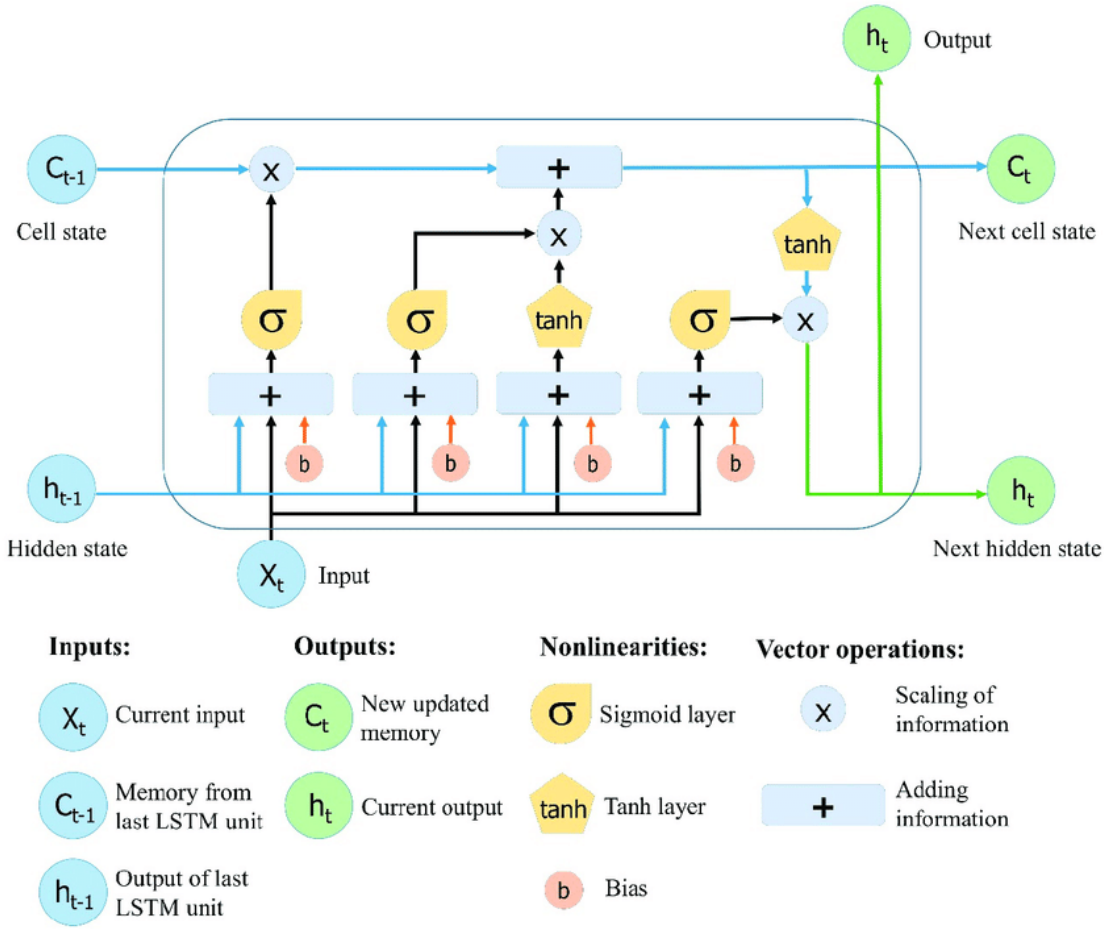
Figura de mai jos prezintă o posibilă structură de RNN. Fiecare stare ascunsă $h(t)$ este pasată către starea următoare $h(t+1)$ împreună cu inputul corespunzător $x(t+1)$. La antrenare input-urile reprezintă câte un cuvânt aferent descrierii. La testare acestea vor reprezenta valoarea predicției anterioare.



În cazul nostru am folsit întâi un strat de scufundare ("Embeding Layer") urmat de o rețea RNN de tip LSTM ("long short-term memory"). Aceasta din urmă este un tip de rețea capabilă să învețe dependențe de lungă durată. O rețea LSTM este formată din "Forget Gate", "Input Gate" și "Output Gate".

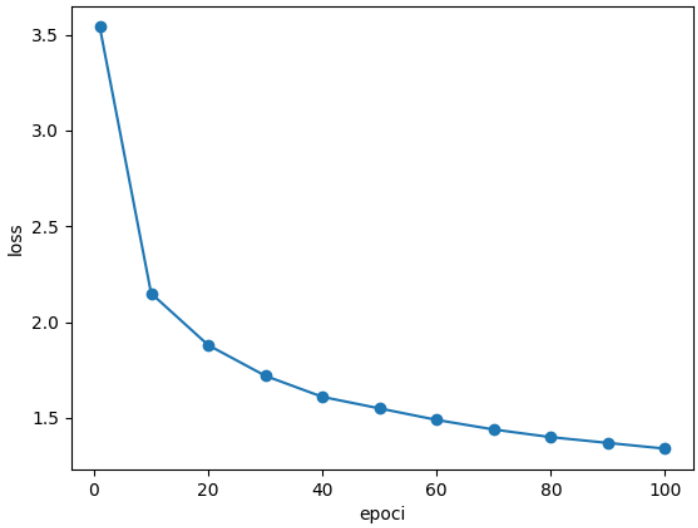
"Forget Gate" decide pe baza stării anterioare și a inputului curent ce informații nu vor mai fi luate în considerare. "Input Gate" este responsabilă pentru construirea stării curente, folosindu-se de un vector de valori candidat și de poarta "Forget Gate". "Outpup Gate" construiește output-ul stării curente pe baza produsului dintre activarea *sigmoid* a sumei dintre starea precedentă și input, precum și activarea *tanh* a output-ului porții "Input

Gate". Comportamentul acestei rețele este ilustrat în figura de mai jos.



Rezultate

Am antrenat modelul 100 de epoci folosind funcția de loss "CrossEntropyLoss", optimizatorul "Adam" și un learning-rate de $5 \cdot 10^{-4}$. Evoluția loss-ului poate fi observată în graficul următor:



Referințe

[1] *Advanced Guide to Inception v3*. URL: https://cloud.google.com/tpu/docs/inception-v3-advanced.

[2] Alex Sherstinsky. "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network". In: *CoRR* abs/1808.03314 (2018). arXiv: 1808.03314. URL: http://arxiv.org/abs/1808.03314.

[3] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. "Show and Tell: A Neural Image Caption Generator". In: *CoRR* abs/1411.4555 (2014). arXiv: 1411.4555. URL: http://arxiv.org/abs/1411.4555.