Підсумовування текстів за допомогою глибоких нейронних мереж

Борух М. І.
Керівник: Музичук Ю. А.

Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
Кафедра обчислювальної математики

12 травня 2021 р.

Зміст

Постановка задачі

Маємо набір текстів $X^{n\times m}$ і їх підсумовані варіанти $Y^{n\times k}$, де n – кількість даних, m – довжина тексту, k – довжина підсумку відповідного запису. Метою тренування мережі є вивчення зв'язків між X та Y, так щоб модель максимізувала ймовірність Y, з урахуванням X:

$$\max p(Y \mid X) = \max \prod_{t=1}^{n} p\left(y_t \mid y_{<t}, X\right),$$

де $y_{<t}$ – основні ознаки попередніх кроків.

Обробка даних

- Токенізація або лексичний аналіз
- Нормалізація
- Видалення ”шуму”

Після застосування попередніх пунктів виконується перетворення тексту у векторне представлення використовуючи ”The Continuous Bag Of Words Model”. Ідея якого перетворити слова у числа так, щоб слова близькі за значенням мали подібне представлення.

LSTM клітина

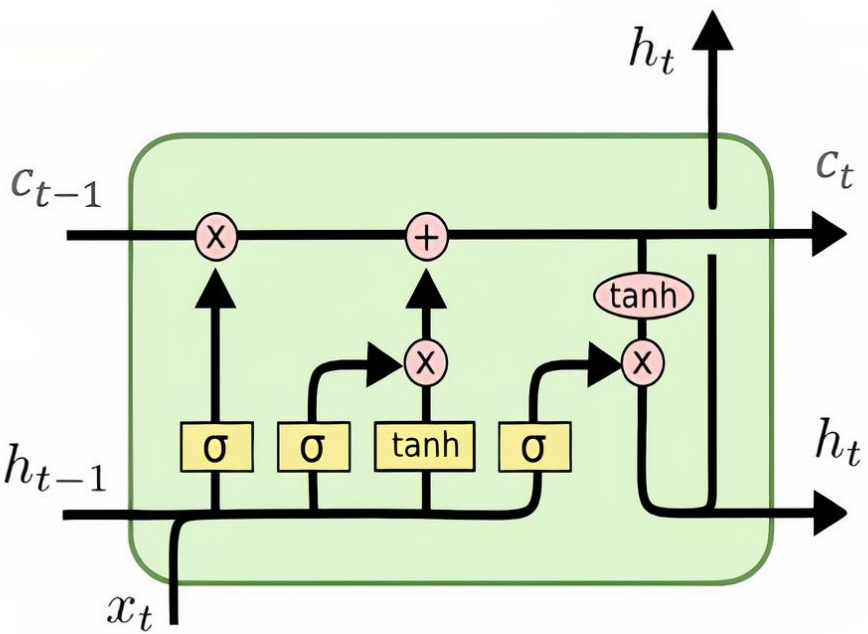

Рис.: LSTM клітина

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right)$$

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right)$$

$$\tilde{C}_t = \tanh\left(W_C \cdot [h_{t-1}, x_t] + b_C\right)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

$$o_t = \sigma\left(W_o [h_{t-1}, x_t] + b_o\right)$$

$$h_t = o_t * \tanh\left(C_t\right)$$

Модель

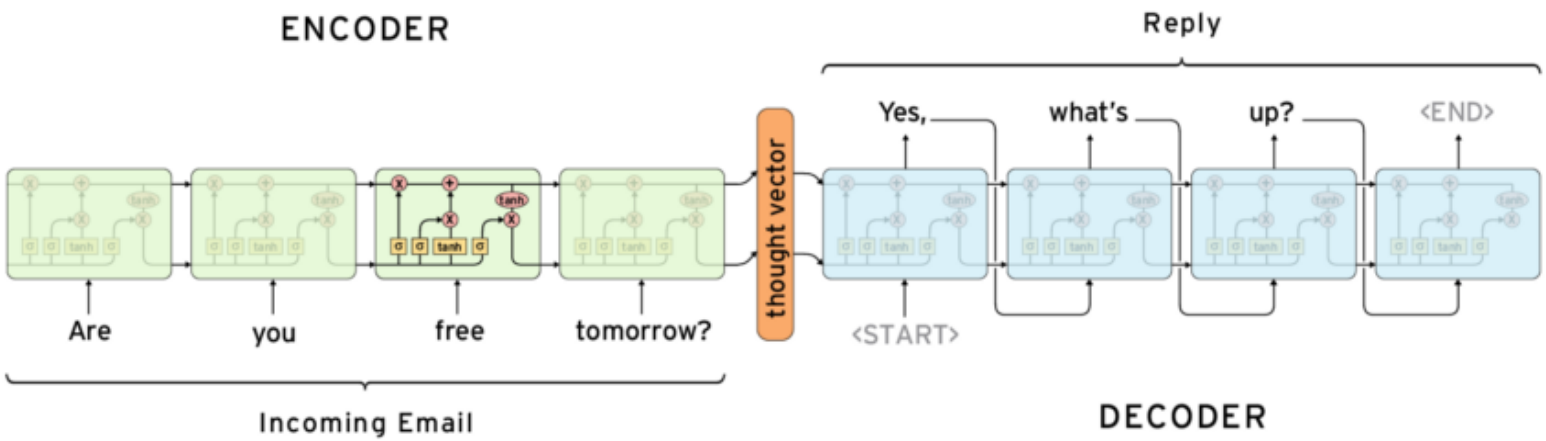

Рис.: seq2seq

Загальна структура

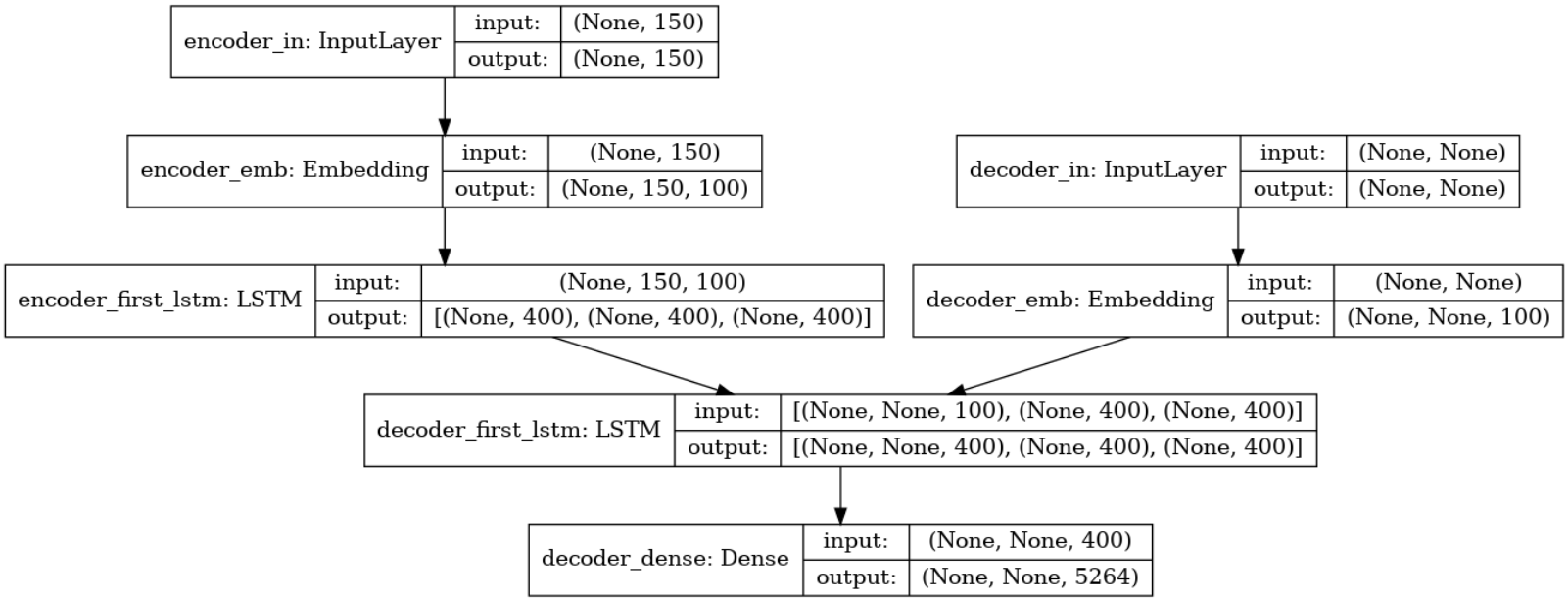

Рис.: Архітектура мережі

Жадібний алгоритм – ідея якого полягає в тому, щоб перебрати всі ймовірності й знайти слово якому відповідає найбільша ймовірність і обрати його.

Променевий пошук повертає список найбільш ймовірних речень.

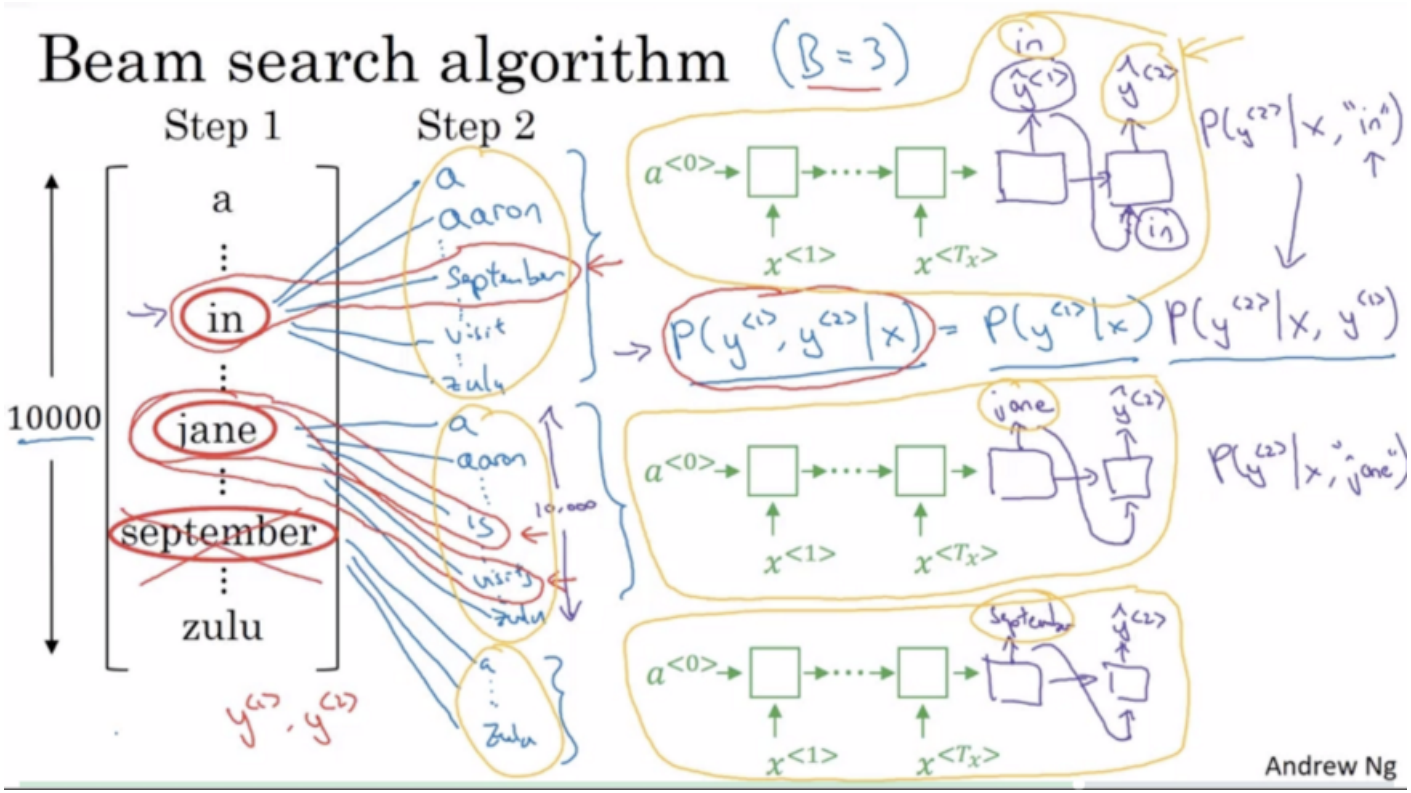

Рис.: Променевий пошук

Топ-k прикладів

Топ-k прикладів – основна ідея в тому, щоб з ймовірностей передбачених мережею зрізати верхні k.
Для найкращих k наступних слів (токенів) формується новий словник слів $V^{(k)} \subset V$. Розподіл масштабується за наступним правилом

$$P'\left(x \mid x_{1:i-1}\right) = \begin{cases} \frac{P(x|x_{1:i-1})}{\sum_{x \in V^{(k)}} P(x|x_{1:i-1})} & \text{якщо } x \in V^{(k)} \\ 0 & \text{інакше.} \end{cases}$$

Наступне слово обирається на основі даного розподілу.

Метод ядер

Метод ядер – основна ідея така сама як в топ-k прикладів з тією відмінністю, що інакше формується словник.
Для найкращих p наступних слів (токенів) формується новий словник слів $V^{(p)} \subset V$ так, щоб задовольнити наступну умову:

$$\sum_{x \in V^{(p)}} P\left(x \mid x_{1:i-1}\right) \geq p$$

Далі так само як і в попережньому методі.

Результати

Оригінальний підсумок: know when to seek medical help do not apply direct heat to the person avoid exposure to cold understand who is at risk for take steps to prevent risk

Згенерований підсумок: talk to your doctor about your medications

		f	p	r
ROUGE-1	Avg	0.144	0.19	0.121
	Max	0.562	0.75	0.473
	Min	0.0	0.0	0.0
ROUGE-L	Avg	0.126	0.176	0.103
	Max	0.471	0.75	0.462
	Min	0.0	0.0	0.0

Таблиця: ROUGE для підсумків

Висновки

Було побудовано мережу яка вміє підсумовувати текст. Для її побудови було використано LSTM шар через його вміння запам'ятовувати довготривалі залежності. Розглянули різні методи відбору слів для формування підсумку, щоб уникнути феномену виродження тексту. Отримані підсумки оцінили з використанням метрики ROUGE. Хоча і метрика ROUGE показала ненайкращі результати, можна стверджувати що дана мережа робить хороші підсумки.

Дякую за увагу