Spell Correction Model
09-03-2024

Document → **Read** → OCR → Optical Character Recognition

Task → spell correction model

Build a DL model to corret the wrong spelling

Price → [DL mode] → price

Pric → [DL model] → price

---

price

price — sequential — LSTM

pric → price

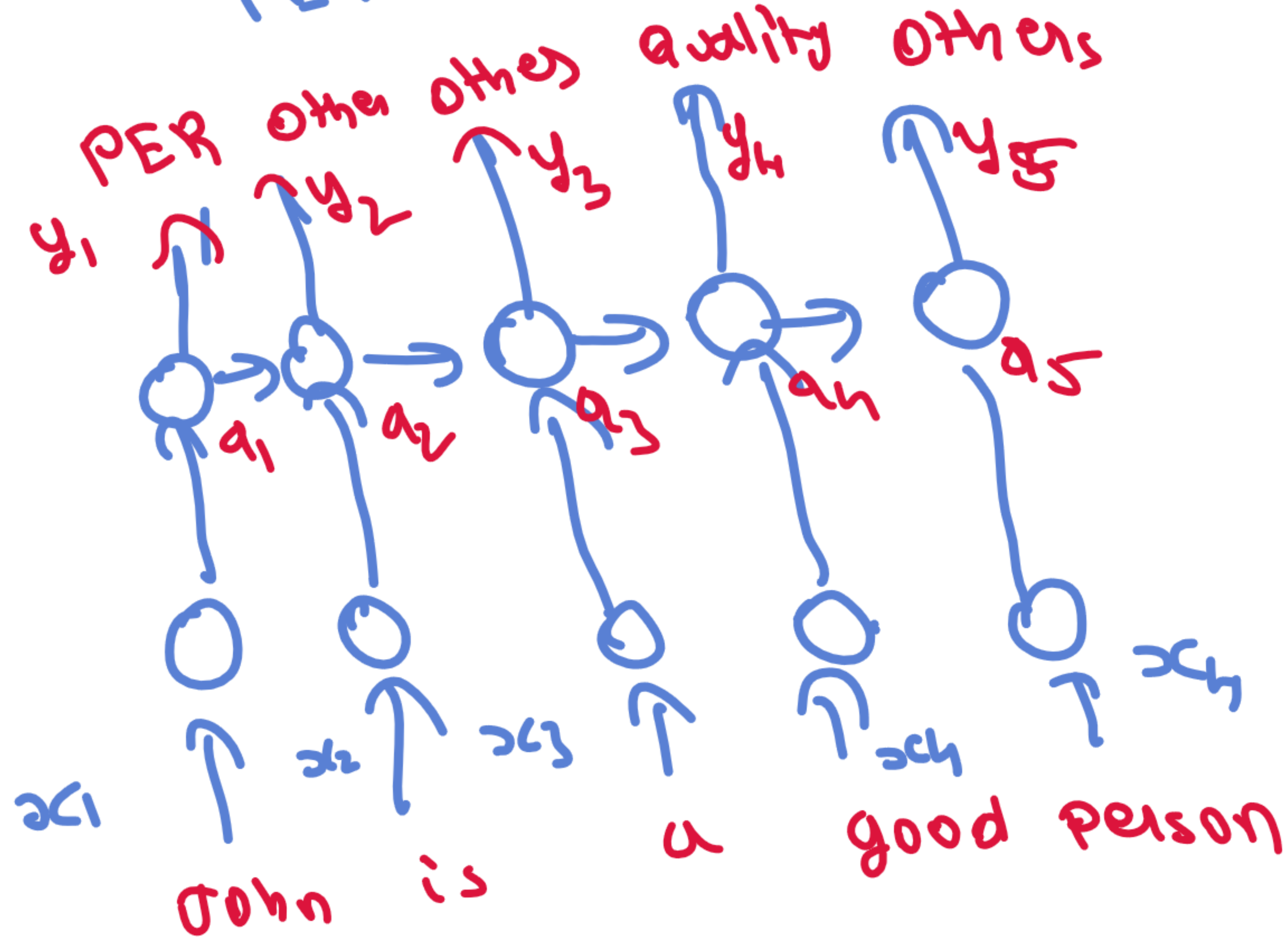
sequence to sequence

(1) len(x) = len(y)

NER → Named Entity Recognition

John is a good person

John → PER; is → oth; a → othe; good → Quality; person → others



(5) ⇒ (5)

② len(x) != len(y)

Spell Correction Model

→ Chatbot (Q&A)
→ Language translation } → words
→ Text summary
→ Spell correction } → each character

len(x) != len(y)
pric ⇒ price

↳

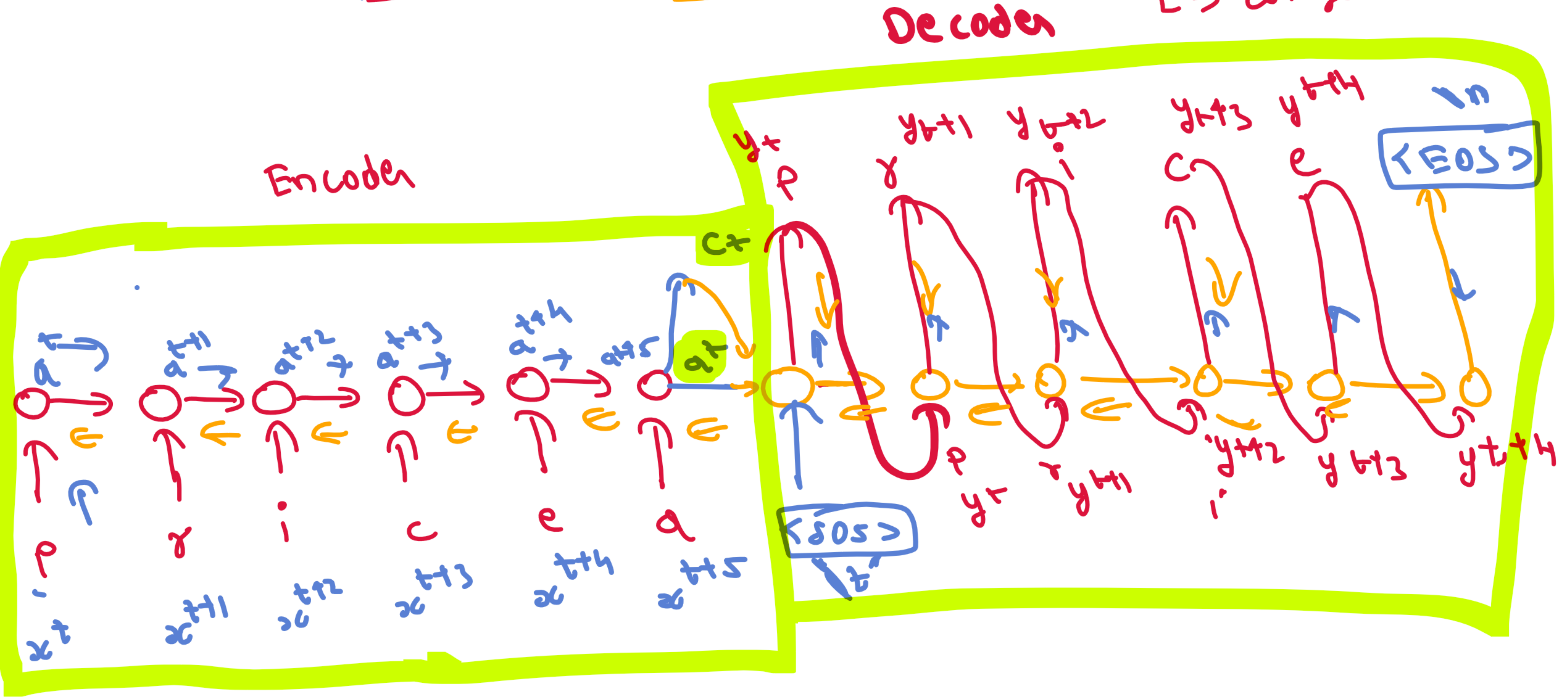
Encoder - Decoder Architecture

→ spell correction → each characte is a input

→ each character is a token

i/p pricea ⇒ o/p price

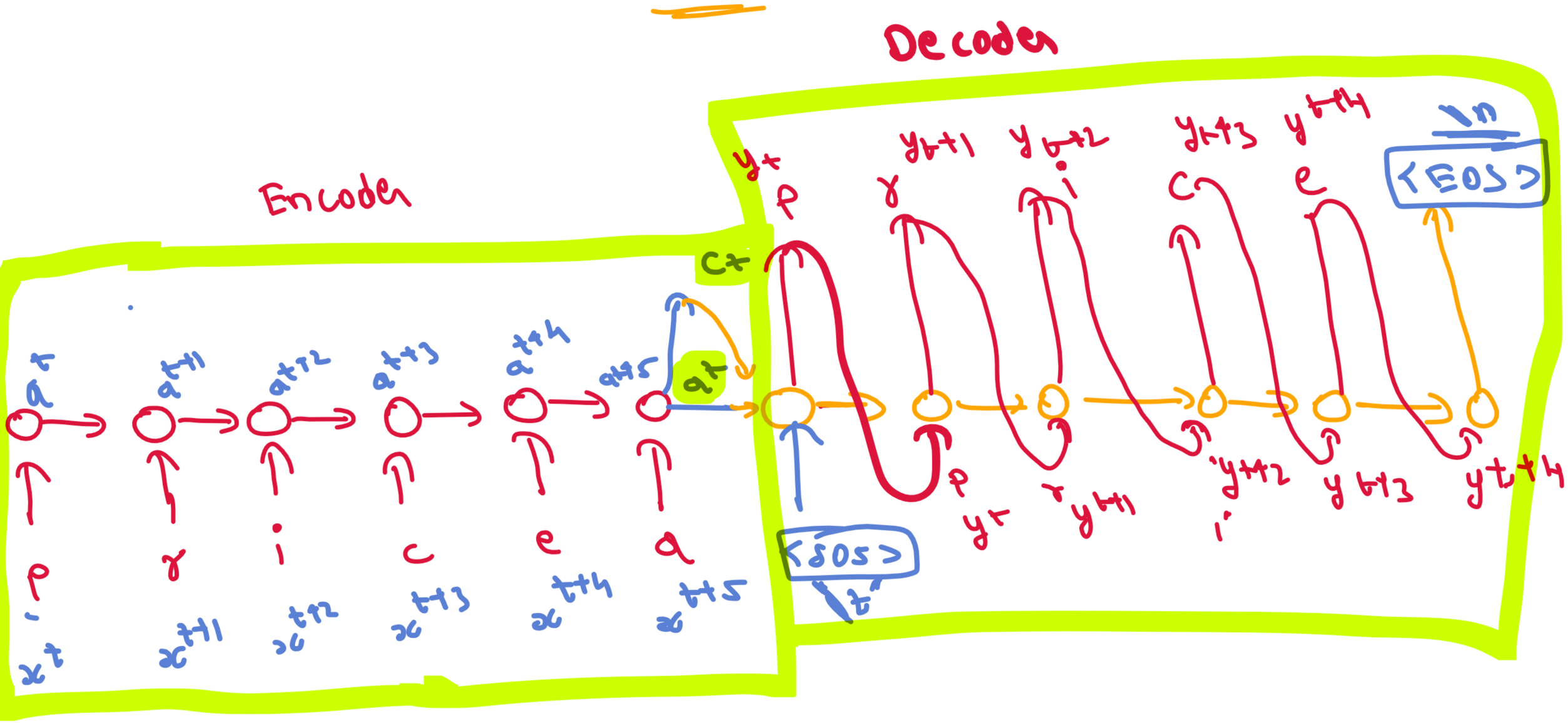
O → LSTM / RNN / GRU
↳ categorical



C_t → cell state - context or memory

h_t or a_t → hidden state → previous activation output

SOS → start of sentence ('\t')



c_t → cell state — context & memory

h_t and a_t → hidden state → previous activation output

SOS → start of sentence ('st)

EOS → End of sentence ('n)

Encoder

① Encode will take the user i/p & understant that contex & store in c_t & a_t

Encoding → Encoding the input in a_t, c_t

Decoder

② Decoder receive the states from encoder

③ Given the states the Decoder learn to predict the output

Deco → max - token limit ⇒ 150

https://tinyurl.com/spell-correction-data

wow — ②

plac — ③

Index word pos: wor

word2 Index word: pos

→ Character missing → information → inormation

→ Extra character → price → pricaea

→ wrong character → degra → deta

Double character ⇒ deeta

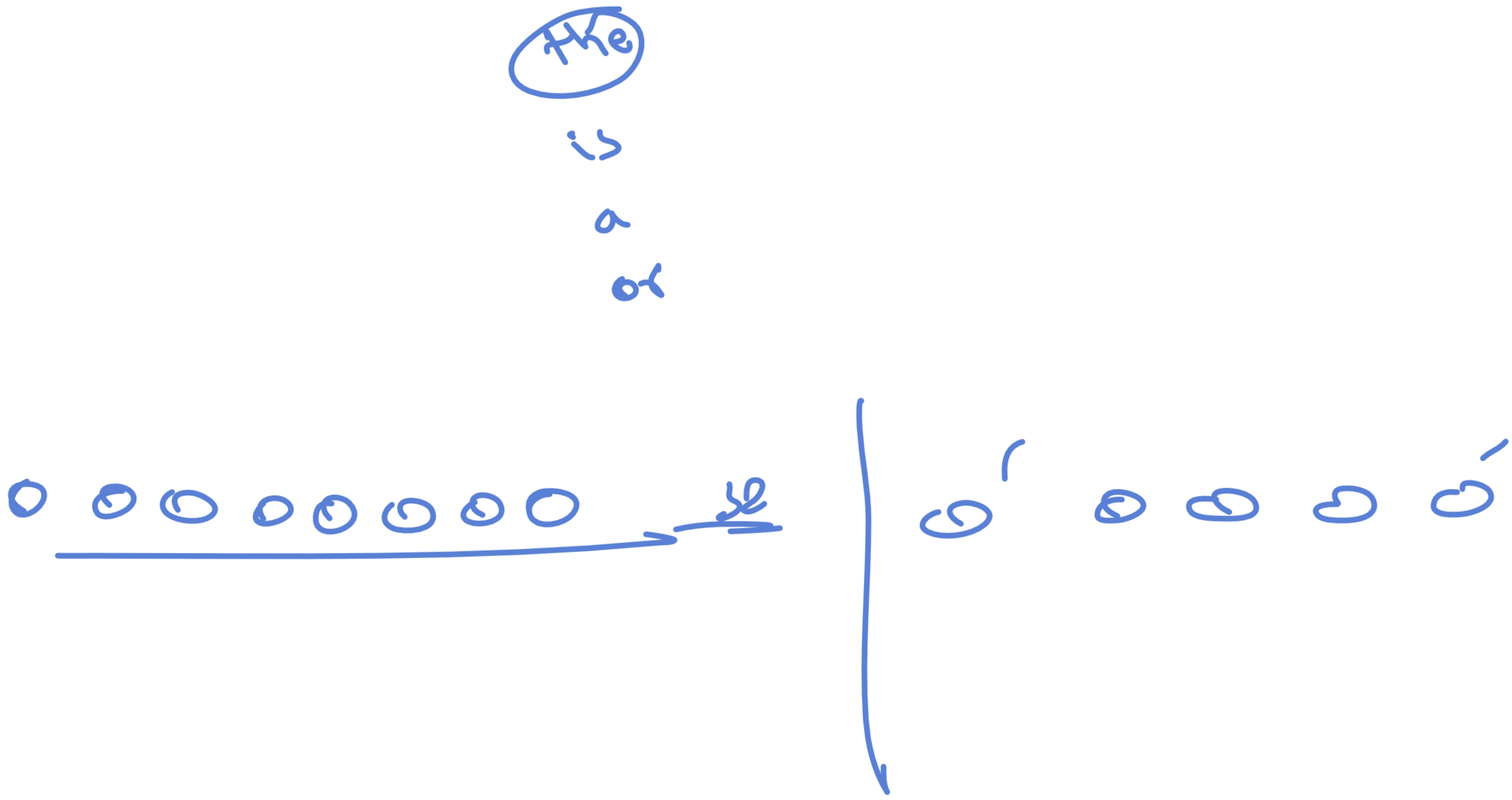
order

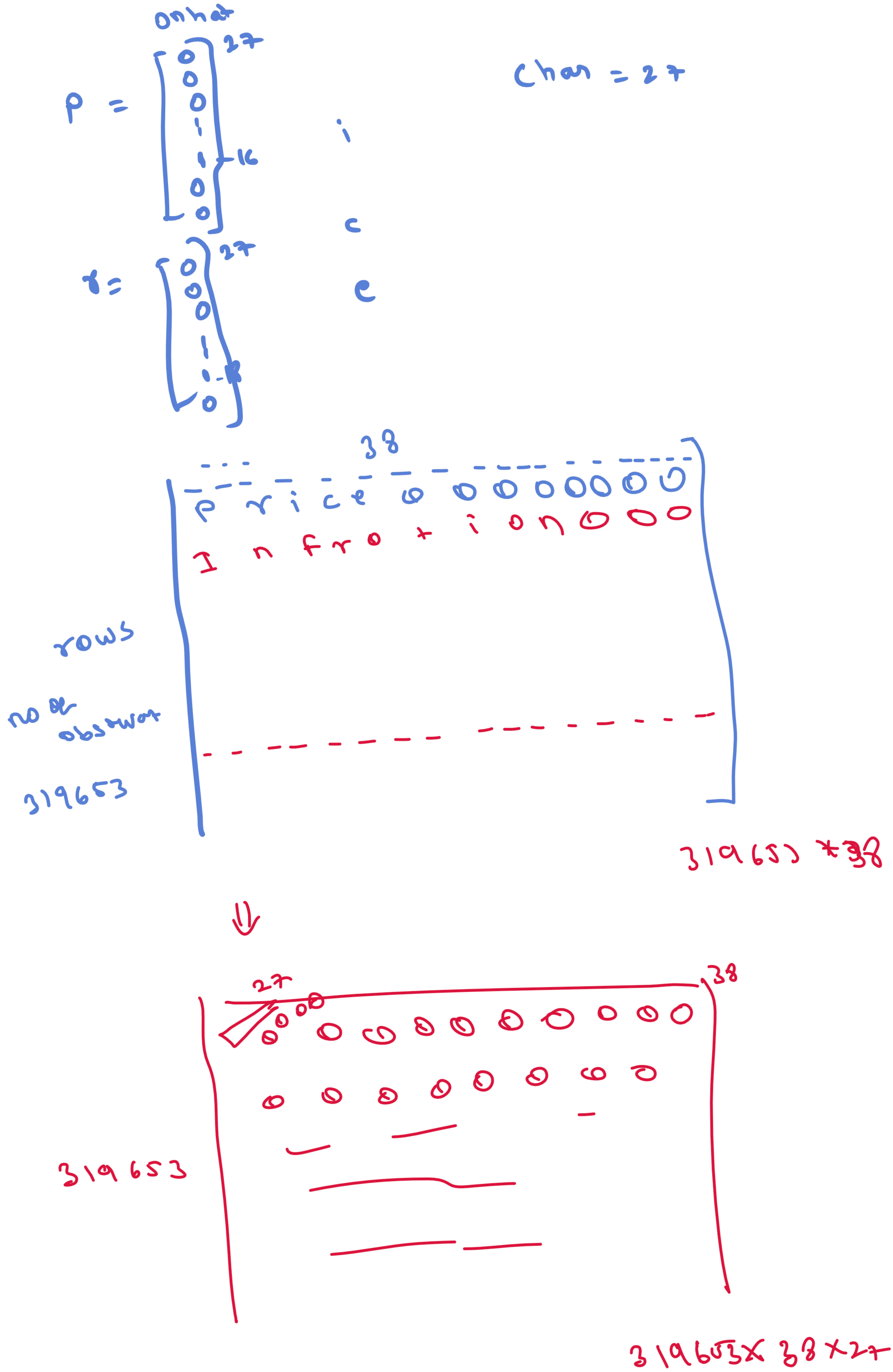
!

price → price

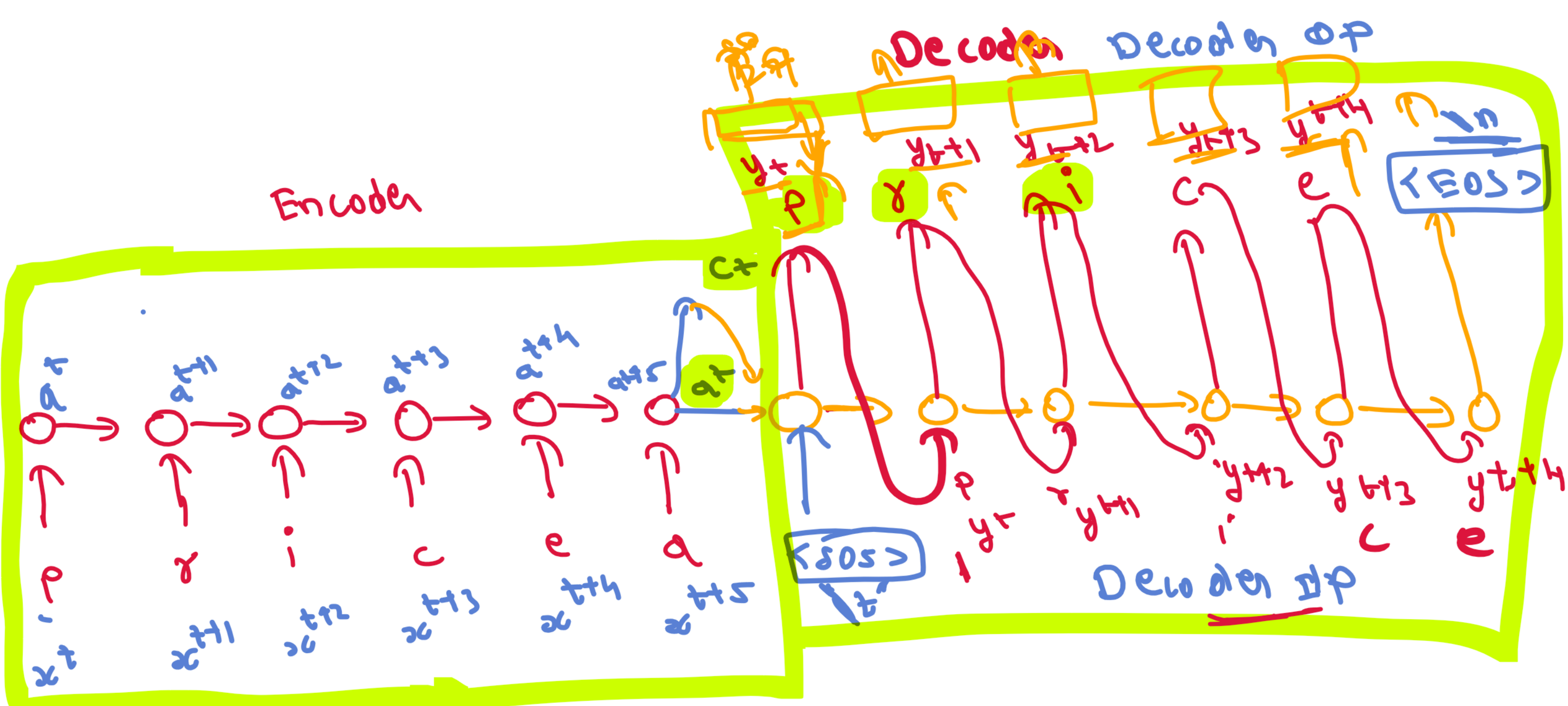
one → one

core — correct





onhot
27
p =
16
i
c
e
r =
27
Char = 27
38
price
Infrotion
rows
no of observat
319653
319653 * 38
27
38
319653
319653 X 38 X 27



Encoder ip

Encoder ip → price

Decoder op → price <EOS>

Decoder ip → <SOS> price