# Create Input target pairs

o Last step before we create vector embeddings is to create input target pairs

* what do these input-target pairs look like?

text sample

□ is target, before is input

iter 1 LLMs learn

2 LLMs learn to

3 LLMs learn to predict ~~ ← target to predict

← LLMs cannot access words past the target

4 LLMs learn to predict one

5 ~

6 ~

7 ~

8 LLMs learn to predict one word at a time

* Given a text sample
  - ↳ Extract input blocks as subsamples that serve as input to LLM
  - ↳ LLMs prediction task during training is to predict the next word that follows the input block
  - ↳ during training, we mask out all words that are past the target

---

* Data loader that fetches input-output target pairs using a sliding window approach

text: "In the heart of the city stood the old library, a relic from"

tensor input: x = [["In", "the", "heart", "of"],

2

3

4

])

output tensor y = [["the", "heart", "of", "city"]

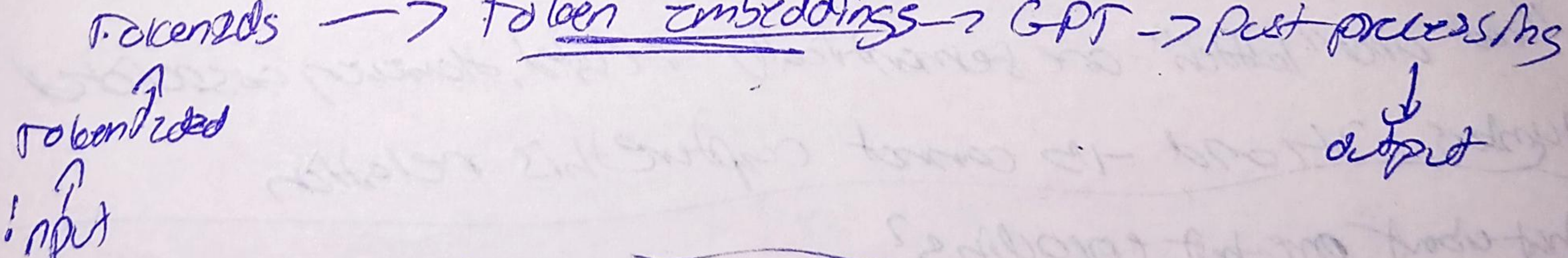
1 2 3 4

]]

# To implement efficient data loaders, we collect inputs in a tensor X where each row represents one input context. the second tensor Y contains corresponding prediction targets (next word), which are created by shifting input by one position.

---

Token Embeddings



① conceptual understanding of why token embeddings are needed

② Demo

③ How are token embeddings created for LLMs

random values