XCS236 PS1

Agenda

- 25 mins written questions
- 25 mins coding
- 10 mins Q&A

Q1 Maximum likelihood estimation and KL divergence

- **Algebraic manipulations. Prove LHS = RHS**
- *MLE over joint distribution p(x,y) = a discriminative model trained over min. KL divergence over p(x) marginal distribution*
- FAQ
  - Expectation subscript = the probability distribution we're using to do the expectation. Weighting factor.
  - LHS relates to generative models because derived from joint empirical dist.
- Tips
  - Expand KL term in RHS
  - During write up, be careful with subscript notations

Q2 Logistic regression and naive bayes

- **Algebraic manipulations. Prove LHS = RHS**
- *Joint probability from GMM with strong independence assumptions is expressive enough to model conditional distribution of multi-class logistic regression model*
- FAQ
  - GMM? many gaussians mixed together form an expressive distribution
  - Naive bayes? Covariance = diagonal. Strong independence assumption between each gaussian
  - Multiclass logistic regression? Softmax to map lin. comb. features into probs.
  - GMMs are much more expressive than logistic regression
- Tips
  - Free: exploit bayes rule to flip conditional dist.
  - Expand terms. Keep simplifying
  - Then equate the terms of GMM parameters to those of multiclass logistic regression model

Q3 Conditional independence and parameterisation

- **Calculation/deduction**
- Tips
  - Draw diagram for yourself for c)

Q4 Autoregressive models

- **Proof by construction. Yes with proof or No with counterexample**
- *Forward and backwards autoregressive factorisation both model the full joint probability. A single model could do both if its expressive enough. But can gaussians?*
- Tips
  - Evaluate the marginal distribution p_f(x_2). What do you observe?

Q5 Monte carlo integration

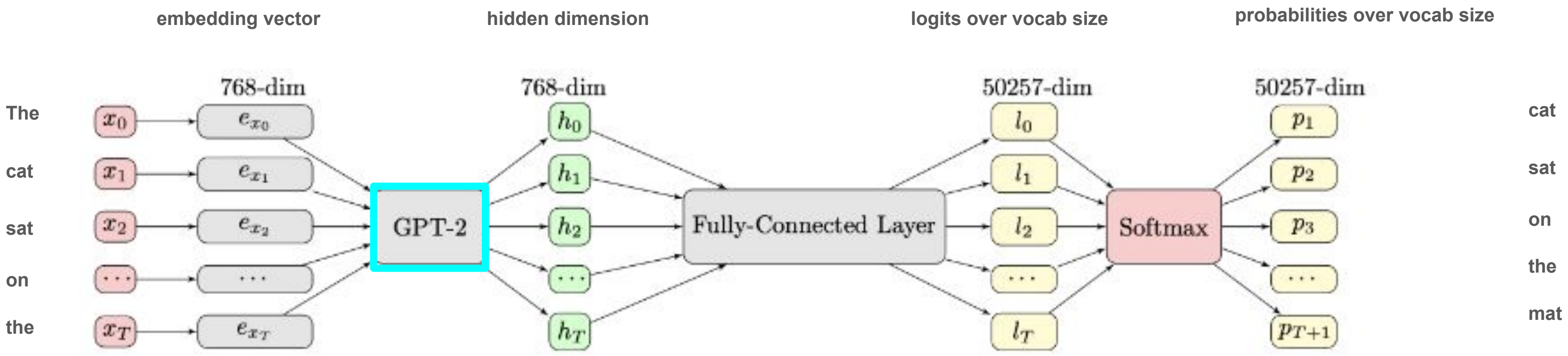
- **Algebraic manipulations to prove LHS = RHS**
- *In high dimensional spaces, exact integration of marginal probability is intractable so we solve analytically via monte carlo integration*
- FAQ
  - So what? In latent generative models, if z is high dimensional it's an insane number of combinations to sum over
  - Want to evaluate p(x) because we need it to do MLE
  - We want to do MLE to fit a generative model to data
  - p(x) is evaluated repeatedly during MLE right? so if p(x) is expensive/intractable so is MLE
- Tips
  - a) take the expectation. Exploit properties of Expectations.
  - b) Jensen's inequality. Be careful with the direction of its inequality. Different for convex vs. concave.

Q6 code setup

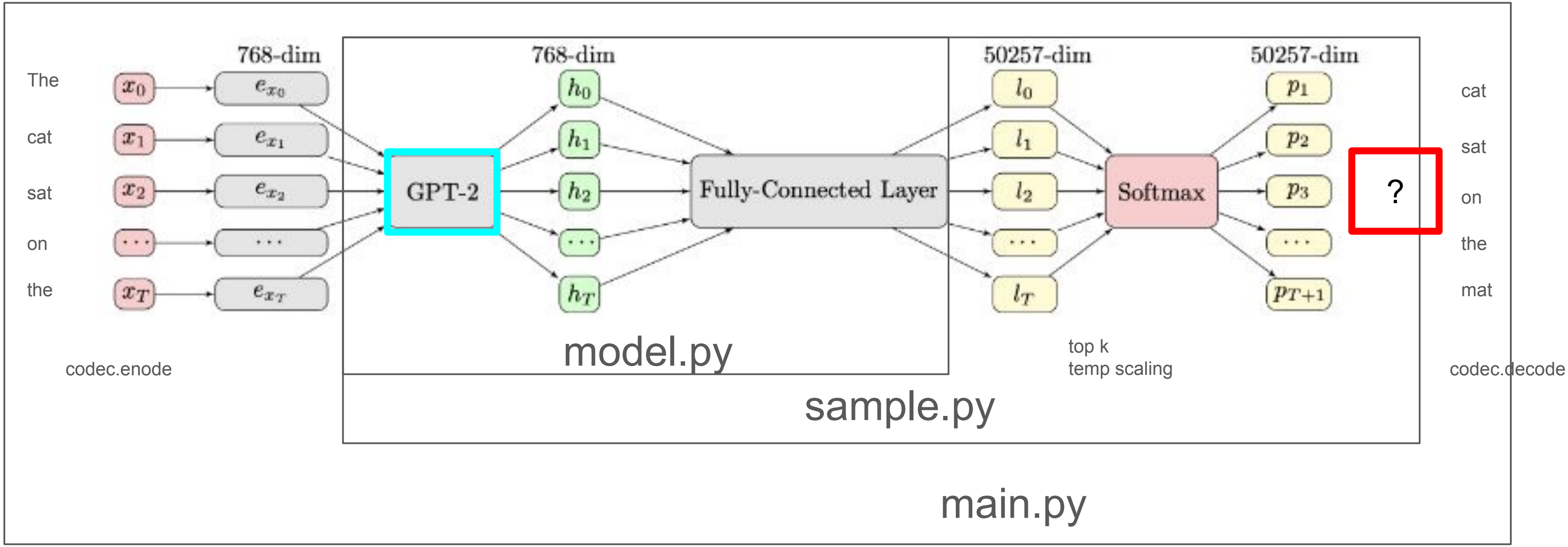
- git clone from https://github.com/scpd-proed/XCS236-PS1
- Install environment.yml or environment_cuda.yml
- run **main.py** first time to download GPT-2 checkpoints
    - Ignore the error in Q6c. It will work when you complete the question
- You don’t need to understand the workings of GPT-2
- But you do need to understand how its inputs and outputs are manipulated

Q6 repo tour

Q6: The big picture



Q6: The big picture



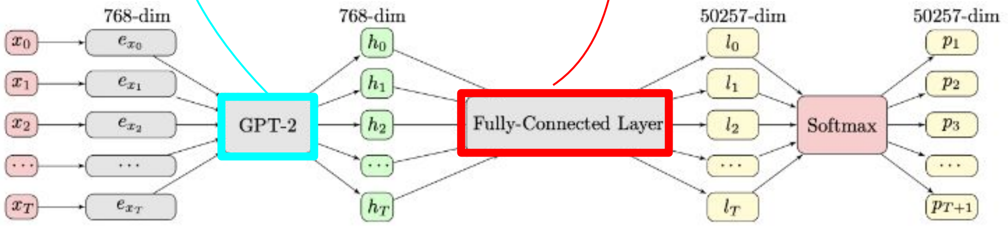
Q6: The big picture

```python
class GPT2(nn.Module):
    def __init__(self, config):
        super().__init__()
        self.transformer = Transformer(config)
        self.readout_head = LinearReadoutHead(self.transformer.wte.weight, config)

    def set_tied(self):
        """ Make sure we are sharing the embeddings
        """
        self.readout_head.set_embeddings_weights(self.transformer.wte.weight)

    def forward(self, input_ids, position_ids=None, token_type_ids=None, past=None):
        hidden_states, presents = self.transformer(input_ids, position_ids, token_type_ids, past)
        logits = self.readout_head(hidden_states)
        return logits, presents
```

PS1 > src > ! config.yml

```yaml
seed: 20190929
vocab_size: 50257
n_positions: 1024
n_ctx: 1024
n_embd: 768
n_layer: 12
n_head: 12
layer_norm_epsilon: 0.00001
initializer_range: 0.02
top_k: 40
```

The cat sat on the



cat sat on the mat

Q6: Implementation tips

- Hints in comments are generous
- Slack help

```
        ##TODO:
        ## 1) sample using the given `logits` tensor;
        ## 2) append the sample to the list `output`;
        ## 3) update `current_text` so that sampling can continue.
        ##    Hint: Checkout Pytorch softmax: https://pytorch.org/docs/stable/generated/torch.nn.functional.softmax.html
        ##                   Pytorch multinomial sampling: https://pytorch.org/docs/stable/generated/torch.multinomial.html
        ## Hint: Implementation should only takes 3~5 lines of code.
        ##       The text generated should look like a technical paper.
        ##
        ## Note: It is expected that the code will throw an error until you've filled out the code block below.
        ### START CODE HERE ###
        ### END CODE HERE ###

        past = new_past

    output = torch.cat(output, dim=1)
    return output
```

Q6: Debugging tips

- <tensor>.shape and <tensor>.size()
- print(<variable>)
- Breakpoints in main.py
- Breakpoints in grader.py

6f Long temperature horizon scaling

$\log p(x) = \sum_i \log p(x_i|x_{<i})$. When sampling with a temperature T, they rescale each univariate conditional by T.

$$\log p_T^{\text{myopic}}(x_i|x_{<i}) = \log \frac{e^{(\log p(x_i|x_{<i}))/T}}{\sum_k e^{(\log p(x_i=k|x_{<i}))/T}} \qquad (2)$$

This approach is efficient since it handles one dimension at a time and only requires rescaling the output logits. However, since the scaling is *myopic*, the chain rule factorization does not preserve the scaled joint distribution in Eq 1.

$$\log p_T(x) \neq \sum_i \log p_T^{\text{myopic}}(x_i|x_{<i}) \qquad (3)$$

It is easy to see that in the extreme case, myopic scaling of an autoregressive model with $T \to 0$ will not necessarily produce the argmax sample of the joint distribution.

cost, LHTS only requires a one
after which long horizon temp
generated directly without sea

Biasing the model towards hi
also be viewed as controlla
vant works include Quark (I
tions the dataset based on a c
toxicity), and reinforces the
tions. Other works on controll
conditional generation, for ex
for images (Nichol & Dhariw

Finally, LHTS relates closely t
man & Goodman, 2014), sinc
intractable temperature-scale
temperature approaches zero
inference (Koller & Friedman

https://www.andyshih.com/assets/pdf/ShihSEicml23.pdf