Last Name: ______________________ **First Name:** ______________________

SUNet ID: ________________________________@stanford.edu

Midterm Solutions — October 24, 2025

Closed-book exam. No notes, no laptop, no phone, no internet, no AI assistance, no anything. Just your brain and a pen!

Duration: 1 hour 30 minutes **Total number of points: 100**

Instructions

Questions are grouped into logical sections to ease your thought process. There are two types of questions:

- **Multiple-choice**: One correct choice per question. Please circle the correct answer.
- **Free-form**: Short and concise answers.

In all cases, there is no penalty for wrong answers.

I. Transformer background and architecture (25 points)

1. (1 point) Relative to word-level tokenization, a key advantage of subword methods (e.g., BPE/WordPiece) is:

   A. They eliminate the need for a vocabulary.

   B. They reduce out-of-vocabulary issues by learning frequent stems/prefixes/suffixes.

   C. They make embeddings fully interpretable without training.

   D. They require no training data.

2. (1 point) In word2vec, which proxy task predicts a masked *center* word from its surrounding context?

   A. Skip-gram

   B. CBOW

   C. MLM

   D. NSP

3. (1 point) A core limitation of vanilla RNNs for long-range dependencies is:

   A. Too many parameters compared to attention

   B. Vanishing/exploding gradients through many sequential multiplications

C. Lack of a hidden state

D. Inability to process character tokens

4. (2 points) Which sublayer appears in the *decoder* stack but not in the *encoder* stack?

   A. Feed-forward network

   B. Self-attention

   C. Encoder–decoder (cross-)attention

   D. Layer normalization

5. (2 points) In scaled dot-product attention, dividing by $\sqrt{d_k}$ primarily prevents:

   A. Overfitting via regularization

   B. Softmax saturation from large dot products

   C. Loss of positional information

   D. Gradient checkpointing overhead

6. (2 points) What is the main benefit of *multi-head* attention?

   A. Eliminates the need for FFN layers

   B. Captures diverse interaction patterns in parallel

   C. Reduces complexity below linear time

   D. Removes the need for positional encodings

7. (2 points) Layer normalization in Transformer blocks primarily:

   A. Normalizes each token's hidden features (per position) to stabilize and speed up training

   B. Normalizes across the batch and time dimensions like BatchNorm

   C. Computes attention scores more efficiently than dot products

   D. Eliminates the need for residual connections

8. (1 point) In decoder self-attention for language modeling, the mask:

   A. Prevents attending to future tokens

   B. Prevents attending to past tokens

   C. Forces uniform attention

   D. Disables attention on special tokens

9. (3+4 points) **Scaled dot-product attention.** (i) Write the core formula for self-attention and (ii) briefly explain the role of Q, K, and V.

> (i) $\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V$
>
> (ii) Q is the query matrix, K is the key matrix, and V is the value matrix. $QK^\top$ is the score matrix, and $\text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)$ is the attention weights. V is then weighted by the attention weights to get the output.

10. (3+3 points) **Label smoothing.** (i) Summarize what label smoothing is optimizing for and (ii) why it helps generalization.

> (i) Label smoothing is a regularization technique that replaces one-hot labels with a smooth distribution over the classes. It helps prevent overfitting by encouraging the model to learn a more nuanced representation of the classes.
>
> (ii) Label smoothing helps generalization by encouraging the model to learn a more nuanced representation of the classes. It prevents the model from overfitting to the training data by encouraging it to learn a more generalizable representation of the classes.

II. Transformer-based models & tricks (25 points)

1. (1 point) A benefit of sinusoidal (hardcoded) positional encodings is that they:
   - A. Require retraining to extend to longer sequences
   - B. Enable length extrapolation without retraining
   - C. Directly encode relative positions via learned per-head bias
   - D. Remove the need for token embeddings

2. (2 points) BERT's pretraining objective primarily uses:
   - A. Masked language modeling (MLM)
   - B. Autoregressive next-token prediction
   - C. Sequence-to-sequence denoising only at the decoder
   - D. Next-sentence generation without masking

3. (2 points) Which architecture is *bidirectional* at pretraining time and well-suited for sentence encoding/classification?
   - A. Decoder-only models (e.g., GPT)
   - B. Encoder–decoder models
   - C. Encoder-only models (e.g., BERT)
   - D. Recurrent neural networks only

4. (2 points) Rotary Position Embeddings (RoPE) primarily:
   A. Add sine/cosine vectors to token embeddings
   B. Mask attention to long-range tokens
   C. Downweight values uniformly with distance
   D. Rotate Q and K by position-dependent 2D blocks to encode relative distances

5. (1 point) Which approach is most commonly used today in LLMs for position information?
   A. RoPE applied inside attention
   B. Learned absolute positional embeddings added to tokens
   C. Sinusoidal PEs added to tokens only
   D. Relative position bias added to values only

6. (2 points) Longformer reduces attention cost primarily via:
   A. Low-rank factorization of attention matrices only
   B. Sliding-window local attention plus optional global tokens
   C. Eliminating keys to save memory
   D. Replacing attention with convolutions

7. (1 point) In causal self-attention, the masking is applied to:
   A. The $QK^\top$ score matrix before softmax
   B. The value matrix V after softmax
   C. The token embeddings before the attention block
   D. The output logits after the final linear layer

8. (1 point) The main purpose of a feedforward network sublayer between attention blocks is to:
   A. Aggregate sequence order
   B. Reduce the number of heads
   C. Replace positional encodings
   D. Provide token-wise nonlinearity and channel mixing

9. (3+2+2 points) **Model families.** Compare encoder-only, encoder–decoder, and decoder-only Transformer families: (i) the typical pretraining objective for each, (ii) one example of model for each, and (iii) what has become the practical default for large language models today.

(i) Encoder-only: Masked language modeling. Encoder–decoder: Span corruption prediction. Decoder-only: Autoregressive next-token prediction.

(ii) Encoder-only: BERT. Encoder–decoder: T5. Decoder-only: GPT.

(iii) Decoder-only.

10. (4+2 points) **RoPE intuition.** (i) How does rotating Q and K make attention depend on *relative* positions? (ii) Name one practical benefit.

(i) Rotating Q and K by position-dependent 2D blocks to encode relative distances makes attention depend on relative positions.

(ii) Natural property that attention between two tokens is a function of the relative distance between the two tokens.

III. Large Language Models (25 points)

1. (1 point) In this course, an LLM is best described as:
   - A. An encoder-only classifier trained with MLM
   - B. A seq2seq model trained with teacher forcing
   - C. A decoder-only autoregressive next-token predictor with masked self-attention
   - D. A biLSTM language model with local attention

2. (1 point) In sparse MoE LLMs, routing works by:
   - A. Activating a learned subset (e.g., top-k experts) per token
   - B. Activating all experts and averaging outputs
   - C. Only using experts during pretraining
   - D. Choosing experts uniformly at random

3. (2 points) PagedAttention primarily aims to:
   - A. Reduce pretraining FLOPs by pruning layers
   - B. Replace attention with convolution
   - C. Train with larger batch sizes by gradient checkpointing
   - D. Manage KV-cache memory with paging to mitigate fragmentation and improve serving efficiency

4. (2 points) Speculative decoding speeds up generation by:

A. Running beam search with a larger beam

B. Using a small draft model to propose tokens that are verified/corrected by the target model

C. Quantizing the target model online

D. Removing the softmax temperature

5. (1 point) The purpose of the KV cache at inference time is to:

   A. Avoid recomputing past keys/values to reduce per-token latency

   B. Store gradients for backpropagation

   C. Save optimizer states between steps

   D. Increase parameter count without extra compute

6. (2 points) Compared to MHA, MQA/GQA reduces latency mainly because it:

   A. Shares Q across heads

   B. Removes caching altogether

   C. Shares K/V across heads (or groups), shrinking the KV cache/bandwidth

   D. Doubles the number of heads

7. (2 points) In nucleus (top-p) sampling, the next token is sampled from:

   A. The top-k tokens by probability

   B. All tokens above a fixed probability threshold τ

   C. The smallest set of tokens with cumulative probability mass $\geq p$

   D. The single highest-probability token

8. (1 point) Increasing softmax temperature T during decoding typically:

   A. Flattens the distribution (more diversity)

   B. Sharpens the distribution (less diversity)

   C. Has no effect on probabilities

   D. Forces beam-search behavior

9. (4+3 points) **Routing collapse.** (i) Define "routing collapse" in sparse MoE training and (ii) give one standard mitigation for it.

> (i) Routing collapse is when the model always routes to the same expert for all tokens.
>
> (ii) Use of an auxiliary loss function to encourage the model to route to other experts as well.

10. (3+2+1 points) **Decoding trade-offs.** Compare greedy/beam search with top-k/top-p sampling in terms of (i) diversity, (ii) quality, and (iii) compute.

> (i) Greedy/beam search: low diversity. Top-k/top-p sampling: more diversity.
>
> (ii) Greedy search: low quality. Beam search: higher quality. Top-k/top-p sampling: higher quality.
>
> (iii) Greedy search: less compute. Beam search: more compute. Top-k/top-p sampling: less compute.

IV. LLM training (25 points)

1. (1 point) The standard pretraining objective for decoder-only LLMs is:
   A. Masked language modeling
   B. Next-sentence prediction
   C. Next-token prediction (autoregressive)
   D. Sequence autoencoding

2. (1 point) Typical pretraining data mixtures in modern LLMs include:
   A. Only supervised instruction–response pairs
   B. Web-scraped text and code at scale
   C. Audio transcriptions only
   D. Purely synthetic tokens

3. (1 point) Supervised finetuning (SFT) is best summarized as:
   A. Freezing the model and adding adapters only
   B. Collecting instruction/response pairs and tuning the model to change behavior
   C. Training a reward model
   D. Doing reinforcement learning with PPO

4. (1 point) Low-Rank Adaptation (LoRA) for parameter-efficient finetuning works by:
   A. Pruning attention heads

B. Quantizing activations

C. Training only embeddings

D. Adding low-rank adapter matrices to the weight updates while keeping the base weights frozen

5. (2 points) Mixed-precision training, as used in practice for LLMs, typically:

A. Stores all tensors in FP64 to avoid numerical error

B. Performs the forward pass in FP64 and the backward pass in INT8

C. Requires changing the model architecture

D. Uses low precision for activations/gradients while keeping a high-precision copy of the weights for updates

6. (2 points) FlashAttention's main optimization is to:

A. Minimize HBM (GPU DRAM) reads/writes via tiling into on-chip SRAM and selective recomputation while keeping attention exact

B. Approximate softmax attention with low-rank projections

C. Replace attention with convolutions to reduce FLOPs

D. Increase sequence length by padding tokens

7. (2 points) In data-parallel training with ZeRO, the variant that shards optimizer state *and* gradients *and* parameters across devices is:

A. ZeRO-1

B. ZeRO-2

C. ZeRO-3

D. None of the above

8. (2 points) QLoRA, as discussed in class, primarily:

A. Stores the frozen base weights in a low-bit format (4-bit NF4 with double quantization) while training small LoRA adapters in higher precision; matrix multiplications are performed in higher precision

B. Quantizes both weights and activations to 1-bit and trains the full model

C. Requires full backpropagation through FP32 copies of all parameters without adapters

D. Only compresses the optimizer state while leaving weights untouched

9. (3+4 points) **Instruction tuning.** (i) Describe what instruction tuning tries to achieve when compared to a pretrained model and (ii) list two practical challenges *discussed in lecture*.

> (i) Instruction tuning tries to achieve better performance on downstream tasks by finetuning a pretrained model on instruction/response pairs.
>
> (ii) 1. Needs high quality data. 2. Sensitive to the choice of the finetuning dataset.

10. (4+2 points) **FlashAttention.** (i) Briefly explain the core idea behind FlashAttention and (ii) state one concrete benefit observed in practice.

> (i) Technique that reframes attention using a combination of tiling and selective recomputation.
>
> (ii) Reduced memory usage and improved inference speed.

*
* *

We hope you enjoyed this exam and the class so far. Looking forward to spending the rest of the quarter together!