The Transformer and GPT: From Encoder–Decoder to Decoder-Only

Lecture Notes Consolidated with Claude AI — CS6140 Machine Learning

---

Instructor's lecture plan for the note of the same title (`TRANSFORMER_claude_1017.pdf`). Section numbers (§) refer to that note.

The note spans one and a half lectures plus a second pass. **Part A** is the second half of Mon 12/7 (after the ATTENTION note's positions and causal mask). **Part B** is the extra lecture, held between 12/7 and 12/10; it is the last lecture HW7 needs. **Thu 12/10** is a workshop: a second pass at the decoder and GPT, the material students have found hardest. If the extra lecture cannot be held in that window, Part B moves to 12/10 and the workshop is dropped. Problem 7's reading is the note's §10 ("Three Kinds of Transformer"), assigned as required reading in Part B with a 5-minute introduction. Mon 12/14 is optional and advanced only (HW7 is due Tue 12/15; the final exam is Thu 12/17). Students arrive knowing scaled dot-product and multi-head attention, masks, positional encoding (ATTENTION note), residual blocks and batch norm (CNN note), and the language model, teacher forcing and BLEU (RNN note).

Part A — Mon 12/7, second half (about 50 minutes): HW7 P4(A–B)

Time	On the board	Worked example / demo
0–5	From the attention translator to "no recurrence at all". The three changes. (§1)	—
5–20	**The big picture**: draw Figure 1 with the three attentions in three colours (as on the 8/8 board) and *leave it up through Part B*. The parts table: point at each box, say where it was seen. (§1)	Table 1: only LN and the wiring are new.
20–30	**The wiring table**: for each attention, queries from / keys and values from / mask. "The decoder asks, the encoder answers." Shapes of the three weight matrices. (§1)	Table 2; (B, h, T, S) for cross-attention.
30–40	Layer norm = batch norm over the other axis; why for sequences (padding, one token at generation, no train/eval). (§2)	$X = [[1, 2, 3], [2, 4, 6]]$: rows $\to (-1.22, 0, 1.22)$, columns $\to (-1, 1)$.
40–50	Add & Norm (the CNN note's residual + LN); the FFN = HW5's MLP per position, "attention communicates, the FFN computes"; the encoder block; where dropout goes. (§3–§4)	Parameter count of one block: 3,280 at $d = 16$; 789,760 at $d = 256$.

Part B — the extra lecture, between 12/7 and 12/10 (100 minutes): HW7 P4(C–D), P5

Time	On the board	Worked example / demo
0–8	Recap from the figure still on the board: encoder block, the wiring table.	—
8–25	**The decoder block**: masked self-attention, cross-attention, FFN. Read the cross-attention line slowly: first argument = queries = English. Why causal here and not in cross-attention. (§5)	The "change the last English word" test.
25–40	**Train in parallel, generate in a loop.** Teacher forcing under the causal mask = one pass; greedy decoding = a loop (draw both rows). The embedding-start box. (§6)	The results table: 27.9 → 37.6 → about 45 BLEU, every bucket.
40–55	**Delete the encoder**: grey out the left half and the cross-attention on the board figure. What remains = encoder block + causal mask = GPT. Same objective as HW6 P3. Learned positions; weight tying. (§7)	Parameter count: 4,530,842 vs GPT-2's 4,526,080; gap 4,762.
55–65	The perplexity ladder, continued from the RNN lecture: 23.8 → 18.6 / 17.5 → 15.6 (HF 14.8). Encoder–decoder vs decoder-only table. (§7)	Optional P8: decoder-only translation at 43.9.
65–76	**Pre-norm vs post-norm**: draw both; unroll pre-norm as $X_0 + \sum f_l$; the ResNet identity path; final LN. "About equal at 4 blocks; it is about depth." (§8)	15.4 vs 15.6.
76–90	**Sampling** by hand: greedy, temperature, top-k on $z = (2, 1, 0, -1)$. Check probabilities, not text. Why greedy repeats. (§9)	The five-row table; **demo** (3 min): stories at $\tau = 0.5/1/1.5$ and top-10.
90–95	**Three kinds of Transformer** (HW7 P7, required reading §10): the three attention-pattern grids; BERT fills in hidden words, T5 fills in hidden spans, GPT predicts the next word.	The grids figure.
95–100	HW7 P4–P5, P7 map; the common bugs list. (§11)	—

Workshop — Thu 12/10 (100 minutes): a second pass at the decoder and GPT

No new material. The goal is that every student can trace one decoder step and one GPT step with shapes, and knows what each HW7 check tests. Run the lecture demo notebook on the projector; students answer before each cell runs.

Time	On the board / at the laptop	Worked example / demo
0–12	Redraw the full Transformer *with the class*: students call out each part and its colour; fill in the wiring table (queries from / keys and values from / mask) for the three attentions. (§1)	Wiring table, filled in live.
12–35	**One decoder step on real tensors**: the shapes of the three attentions' weights, (B, h, S, S), (B, h, T, T), (B, h, T, S); "change the last English word, only the last output moves". (§5)	Demo Segments 2 and 5.
35–55	**Train in parallel vs generate in a loop**, again: one teacher-forced pass vs the greedy loop; what the whole-sequence check and the overfit-one-batch check each prove, and what they don't. (§6)	Demo Segment 6.
55–75	**GPT**: grey out the encoder; the parameter count with tying; pre-norm vs post-norm on the diagram. (§7, §8)	Demo Segments 7–8.
75–88	**Sampling** once more: the $z = (2, 1, 0, -1)$ table; probabilities, not text. (§9)	Demo Segment 9 (HF's warpers).
88–100	HW7 P4–P5 clinic: the common bugs list. (§11)	—

Appendix — Mon 12/14: optional and advanced only (HW7 is due the next day)

Time	On the board	Worked example / demo
0–60	**Tour of TRANSFORMER_ADVANCED**: BERT and T5 in detail; RoPE as the 8/8 rotation board applied to q, k; the KV cache and why memory, not compute, limits serving; beam search; where the compute goes (Td^2 vs T^2d).	RoPE: same offset, same score; cache size of GPT-2 small.
60–100	Questions (HW7 questions welcome; no new required material), and the bridge to an LLM course: tokenizer → embeddings → a stack of pre-norm blocks → next-token loss → sampling.	—

Board layout suggestion. Draw the full Transformer (Figure 1) large, in the middle, in Part A, and never erase it: Part B's GPT is made by greying out half of *that* drawing. Keep the wiring table in a corner; every new sub-layer gets checked against it.

The story, in six beats

The whole thing is one sentence stretched out: *you already have every part; the Transformer is a wiring diagram, and GPT is half of it.*

1. **The last recurrence goes.** Attention already let the decoder look back at every Spanish word. What was left sequential was the GRU itself. Replace it with attention between the words of one sentence, and nothing has to run one word at a time during training.

2. **Three attentions, one operation.** Encoder self-attention, decoder masked self-attention, cross-attention: the same $\text{softmax}(QK^\top/\sqrt{d})V$, with different sources for Q, K, V and different masks. If students can fill in the wiring table, they understand the architecture.

3. **Two known tricks keep the stack trainable.** The residual connection from HW6's ResNet, and a normalization: batch norm turned 90 degrees, so it works for one token, any length, any batch.

4. **Parallel training is what made it win.** Teacher forcing plus the causal mask means every position's prediction is computed in one pass. Generation is still a loop; training is not.

5. **Delete half the figure and you have GPT.** No source sentence, so no encoder and no cross-attention. What remains is the encoder block with the causal mask, trained on HW6's language-model task, and it beats every recurrent model on the same data.

6. **The choices that matter at scale.** Pre-norm keeps a clean identity path through a deep stack; sampling turns probabilities into text; one objective on raw text, every token a label, prompts as the interface: that is why the decoder-only design took over.

If you remember one line: *GPT is the Transformer's encoder block with a causal mask, trained to predict the next word.*