CIS 6200: Emerging Theory of LLMs **Fall 2026**

Topic 4B: Learning with Chain of Thought

Last updated: October 13, 2026 *Surbhi Goel*

Acknowledgements: *The sample-complexity analysis follows Altabaa, Montasser, and Lafferty [1]. The computational results use Klivans and Sherstov [4] and Joshi et al. [5].*

AI Disclosure: *These notes were drafted and revised with AI assistance. If you find an error, please let me know.*

In the previous lecture, we constructed models that compute by writing intermediate results. We now compare learning from answers with learning from worked solutions that include these results: how much data and computation does each require?

1 Learning from answers and CoT

A *chain of thought (CoT)* is the sequence of intermediate values $z = (z_1, \ldots, z_k)$ produced before the final answer y. These may be bits or states of a computation, such as an arrangement of items.

We first assume deterministic, noiseless supervision. Following Altabaa et al. [1, Section 2], each hypothesis h specifies an answer and a CoT for every input x:

$$h(x) = (y, z), \qquad h^{\text{ans}}(x) = y.$$

We call (y, z) the complete output; h^{ans} retains only the answer. Let $\mathcal{H}$ be a finite set of these candidate functions. The learner knows the allowed rules, but not which one is the teacher $h_\star$. We assume $h_\star \in \mathcal{H}$, called *realizability*.

Let $\mathcal{D}$ be a distribution over input problems. Both learners use the same class $\mathcal{H}$ and receive m independent inputs $x_1, \ldots, x_m \sim \mathcal{D}$, labeled by either

$$\begin{aligned} &\text{answer supervision:} && (x_i, h_\star^{\text{ans}}(x_i)), \\ &\text{CoT supervision:} && (x_i, h_\star(x_i)). \end{aligned}$$

Here m counts input problems, not individual steps.

At test time, both learners receive only a new input $x \sim \mathcal{D}$ and may generate their own intermediate values. We judge the final answer:

$$\text{err}_{h_\star}(h) = \mathbb{P}_{x \sim \mathcal{D}}[h^{\text{ans}}(x) \neq h_\star^{\text{ans}}(x)].$$

With the teacher fixed, write $\text{err}(h) = \text{err}_{h_\star}(h)$. For an allowed answer error $0 < \varepsilon < 1$ and failure probability $0 < \delta < 1$, the goal is to return $\widehat{h}$ with $\text{err}(\widehat{h}) \leq \varepsilon$ with probability at least $1 - \delta$ over the training sample and the learner's randomness.

A computation is **answer-consistent** if it matches every observed answer. It is **CoT-consistent** if it matches both the answer and the CoT in every training example. CoT consistency implies answer consistency.

A **consistent learner** returns any candidate $\widehat{h} \in \mathcal{H}$ that matches all the training labels it receives. For example, it can check candidates one by one and return the first consistent one. Realizability guarantees that the teacher is consistent. We ask how many examples make every consistent candidate accurate on new inputs. For now, we allow unlimited computation to search the class.

2 Learning from fewer examples

The following examples show how intermediate outputs distinguish computations that give the same final answer. We then bound the number of CoT-labeled examples needed for learning, and use the cutoff example to prove that answers alone can require more data.

2.1 Two examples

Learning a score cutoff. Suppose a program admits a student only if their score meets a cutoff and they have completed a prerequisite. The learner knows that the cutoff θ is either 60 or 80, but not which. For an input $x = (s, c)$, where s is the score and $c = 1$ if the prerequisite is complete and $c = 0$ otherwise, the teacher computes

$$z = \mathbf{1}\{s \geq \theta\}, \qquad y = z \wedge c.$$

Here $z = 1$ means the score meets the cutoff, and $y = 1$ means both checks pass and the student is admitted. The intermediate check z is the CoT, and y is the final decision. Write $h_\theta(x) = (y, z)$, so $\mathcal{H} = \{h_{60}, h_{80}\}$.

A student with score 70 and a missing prerequisite is rejected under either cutoff. The score check, however, passes under cutoff 60 and fails under cutoff 80. Suppose our data contains only scores 70 and 90, and few of the score-70 students have completed the prerequisite. Admission decisions distinguish the cutoffs only on those few students. The score checks distinguish them on every score-70 student.

Learning the moves. Recall the five-item rearrangements from the previous lecture. Now suppose the learner does not know what the move tokens mean. Each candidate assigns one fixed rearrangement of positions to each token. Starting from *abcde*, we apply the moves in the input sequence. The answer is the final arrangement, and the CoT records the intermediate arrangements, as in the state sequences considered by Altabaa et al. [1, Appendix B, Example 4].

For example, a candidate might interpret a token as "do nothing" when it means "swap positions $1, 2$." Two consecutive occurrences give the same final arrangement under either rule, but the first intermediate arrangement distinguishes them.

2.2 Ruling out inaccurate computations

We now analyze CoT supervision for an arbitrary finite class $\mathcal{H}$. Fix the target $h_\star$ and input distribution $\mathcal{D}$. For each candidate $h \in \mathcal{H}$, let

$$q_h = \mathbb{P}_{x \sim \mathcal{D}}[h(x) = h_\star(x)].$$

This is the probability that h matches the teacher's complete output on one input. Since the training inputs are independent,

$$\mathbb{P}[h \text{ is CoT-consistent}] = q_h^m. \tag{1}$$

Following Altabaa et al. [1], define the *CoT information* of h by

$$\mathcal{I}(h) = -\ln q_h, \qquad q_h^m = e^{-m\mathcal{I}(h)}.$$

Thus $\mathcal{I}(h)$ measures how quickly more examples rule out h. When q_h is close to one, $-\ln q_h \approx 1 - q_h$. In this case, the information is approximately the probability of observing an answer or CoT on

which h disagrees with the teacher. If $q_h = 1$, no training example can distinguish them and the information is zero. If $q_h = 0$, one example rules out h with probability one, and we set $\mathcal{I}(h) = +\infty$.

Call a candidate *inaccurate* if $\mathrm{err}(h) > \varepsilon$. To guarantee that any CoT-consistent learner succeeds, we need to rule out all inaccurate candidates. Candidates with error at most ε are acceptable even if their CoTs differ from the teacher's. Define the smallest information among inaccurate candidates:

$$\mathcal{I}(\varepsilon) = \min_{\substack{h \in \mathcal{H} \\ \mathrm{err}(h) > \varepsilon}} \mathcal{I}(h).$$

We leave the fixed distribution, target, and class out of the notation. If no candidate has error greater than ε, set $\mathcal{I}(\varepsilon) = +\infty$.

Interpreting $\mathcal{I}(\varepsilon)$. Increasing ε removes candidates from the minimum, so $\mathcal{I}(\varepsilon)$ can only increase. Revealing more intermediate labels leaves answer errors unchanged and can only decrease q_h, so the information cannot decrease. Since complete outputs include the answer,

$$q_h \le 1 - \mathrm{err}(h), \qquad \mathcal{I}(\varepsilon) \ge -\ln(1-\varepsilon) \ge \varepsilon.$$

When CoT adds no information. Suppose every candidate uses an empty CoT, the same fixed padding, or a copy of its answer. Whenever two candidates agree on the answer, they also agree on the CoT. In this case,

$$q_h = 1 - \mathrm{err}(h), \qquad \mathcal{I}(h) = -\ln(1 - \mathrm{err}(h)).$$

This is exactly the information available from answers alone.

To rule out all inaccurate candidates, we take a union bound over the class.

Theorem 2.1 (Altabaa et al., finite hypothesis class)**.** *Let $0 < \varepsilon, \delta < 1$. Suppose $\mathcal{H}$ is finite and realizable. If $0 < \mathcal{I}(\varepsilon) < \infty$, then*

$$m \;\ge\; \frac{\ln|\mathcal{H}| + \ln(1/\delta)}{\mathcal{I}(\varepsilon)}$$

ensures, with probability at least $1 - \delta$, that every CoT-consistent hypothesis in $\mathcal{H}$ has answer error at most ε. If there are no inaccurate hypotheses, no examples are needed. If there are inaccurate hypotheses but $\mathcal{I}(\varepsilon) = +\infty$, one example suffices with probability one.

Proof. By (1) and a union bound over inaccurate candidates,

$$\begin{aligned} &\mathbb{P}[\text{an inaccurate hypothesis is CoT-consistent}] \\ &\le \sum_{\substack{h \in \mathcal{H} \\ \mathrm{err}(h) > \varepsilon}} e^{-m\mathcal{I}(h)} \le |\mathcal{H}| e^{-m\mathcal{I}(\varepsilon)} \le \delta. \end{aligned}$$

The last inequality uses the stated sample bound. Outside this failure event, every CoT-consistent hypothesis has error at most ε. Realizability ensures that at least one exists.

If every hypothesis is accurate, the conclusion already holds. Otherwise, if $\mathcal{I}(\varepsilon) = +\infty$, every inaccurate hypothesis has $q_h = 0$. A finite union of zero-probability survival events has probability zero after one example. □

Using $\mathcal{I}(\varepsilon) \ge \varepsilon$, the theorem recovers the usual sufficient sample bound

$$m \ge \frac{\ln|\mathcal{H}| + \ln(1/\delta)}{\varepsilon}. \tag{2}$$

This includes answer-only learning by taking the CoT to be empty.

Information in the cutoff example. To make the comparison precise, fix $0 < \rho \leq 1/4$ and use the following input distribution. The last two columns show the score check followed by the admission decision.

Score	Prerequisite	Probability	Cutoff 60: $z \to y$	Cutoff 80: $z \to y$
90	Complete	$1/2$	$1 \to 1$	$1 \to 1$
70	Missing	$1/2 - \rho$	$1 \to 0$	$0 \to 0$
70	Complete	ρ	$1 \to 1$	$0 \to 0$

The answers differ only in the last row, while the score checks differ in both score-70 rows. Under either target, the other cutoff therefore has answer error ρ and $q_h = 1/2$. For $\varepsilon < \rho$, this gives $\mathcal{I}(\varepsilon) = \ln 2$. Without the score check, the other cutoff has $q_h = 1 - \rho$, giving $\mathcal{I}(\varepsilon) = -\ln(1-\rho) \approx \rho$ for small ρ. If $\varepsilon \geq \rho$, neither candidate is inaccurate and $\mathcal{I}(\varepsilon) = +\infty$ in both cases.

Information in the move example. Use the full vocabulary of $5! = 120$ moves from the previous lecture, with their token meanings unknown to the learner. Let each input consist of $T \geq 2$ tokens chosen independently and uniformly. Write π_t for the arrangement after t moves, with $\pi_0 = abcde$. The answer is $y = \pi_T$, and the CoT is $z = (\pi_1, \ldots, \pi_{T-1})$.

Any inaccurate candidate assigns the wrong rearrangement to at least one token. If the input contains a token whose meaning is wrong, consider the first such occurrence. The candidate and teacher have the same arrangement before this move and different arrangements after it: two different rearrangements of five distinct items cannot give the same result. Thus, matching the complete output requires avoiding every token on which their rules differ.

Fix one token whose meaning the candidate gets wrong. A uniform draw avoids this token with probability $119/120$. Since the T draws are independent, the probability of avoiding it throughout the input is $(119/120)^T$. The candidate may get other tokens wrong as well, so this is an upper bound on its agreement probability:

$$q_h \leq \left(\frac{119}{120}\right)^T, \qquad \mathcal{I}(\varepsilon) \geq -T \ln\left(\frac{119}{120}\right) \geq \frac{T}{120}.$$

Longer sequences give more chances to observe an incorrectly specified move. They also require more intermediate labels per problem.

2.3 Why answers can require more examples

The upper bounds analyze consistent learners. To prove a separation, we must show that *every* answer-only learner needs more examples. If the sample contains no input where two target answers differ, the learner sees identical data under both teachers. The following lemma, adapted from Altabaa et al. [1], bounds the failure probability caused by this ambiguity.

Lemma 2.2 (Answer-only lower bound for two targets)**.** *Suppose two computations h_0, h_1 disagree on their answers with probability ρ:*

$$\mathbb{P}_{x \sim \mathcal{D}}[h_0^{\mathrm{ans}}(x) \neq h_1^{\mathrm{ans}}(x)] = \rho.$$

Let $0 < \varepsilon < \rho/2$. For any learner using $m \geq 1$ independent answer-labeled examples, there is a target $h_\star \in \{h_0, h_1\}$ for which

$$\mathbb{P}[\mathrm{err}_{h_\star}(\widehat{h}) > \varepsilon] \geq \frac{(1-\rho)^m}{2}.$$

The probability is over the training inputs and any randomness used by the learner. The learner may return a predictor outside $\mathcal{H}$; its error is still the probability that its final answer is wrong.

Proof. Choose the teacher uniformly from $\{h_0, h_1\}$, independently of the training inputs and the learner's randomness. Fix any sample of inputs on which the two teachers' answers agree, and fix the learner's random choices. The learner then receives the same data and returns the same $\widehat{h}$ for either teacher.

This prediction rule cannot be accurate for both teachers, since

$$\mathrm{err}_{h_0}(\widehat{h}) + \mathrm{err}_{h_1}(\widehat{h}) \geq \rho > 2\varepsilon.$$

The first inequality holds because wherever the teachers disagree, the prediction must be wrong for at least one of them. Thus at least one of the two equally likely teacher choices makes the learner fail.

All m sampled inputs have agreeing answers with probability $(1-\rho)^m$. Averaging over inputs, learner randomness, and the teacher choice therefore gives failure probability at least $(1-\rho)^m/2$. This is the average failure probability for the two fixed teachers, so at least one has failure probability this large. □

The separation. For the cutoff example, fix $0 < \varepsilon < \rho/2$ and $0 < \delta < 1/4$. The lemma shows that any answer-only learner that succeeds with probability $1-\delta$ for *each* of the two possible targets must satisfy

$$\frac{(1-\rho)^m}{2} \leq \delta, \qquad \text{hence} \qquad m \geq \frac{\ln(1/(2\delta))}{-\ln(1-\rho)}.$$

Since $-\ln(1-\rho) = \Theta(\rho)$ for $0 < \rho \leq 1/4$, this requires $\Omega\big(\rho^{-1}\ln(1/\delta)\big)$ answer-labeled examples.

For CoT supervision, substituting $\mathcal{I}(\varepsilon) = \ln 2$ and $|\mathcal{H}| = 2$ into Theorem 2.1 shows that

$$m \geq \frac{\ln 2 + \ln(1/\delta)}{\ln 2} \tag{3}$$

examples suffice for either target. We have therefore proved

Supervision	Sample-complexity bound
Answers only	$\Omega\big(\rho^{-1}\ln(1/\delta)\big)$ necessary
CoT supervision	$O(\ln(1/\delta))$ sufficient

Note that as ρ shrinks, the condition $\varepsilon < \rho/2$ asks for correspondingly smaller prediction error. This is a statistical separation: there are only two candidates, so finding a consistent one is easy with either supervision scheme.

2.4 Stochastic CoTs and Hellinger distance

Suppose the score-check bit in our example is flipped with probability 1/10 before computing $y = z \wedge c$ from this noisy check. A student with score 70 and a missing prerequisite is still rejected under either cutoff. But the intermediate check is 1 with probability 0.9 under cutoff 60 and 0.1 under cutoff 80. Either teacher can produce either bit, so one observation no longer rules out a candidate. We compare how much probability each candidate assigns to the observed data.

Now let each h in a finite class $\mathcal{H}$ specify a discrete distribution $p_h(y, z \mid x)$, with the target $h_\star \in \mathcal{H}$. The prediction goal is to learn its answer distribution $p_{h_\star}(y \mid x) = \sum_z p_{h_\star}(y, z \mid x)$. A

training example consists of $X_i \sim \mathcal{D}$ and an output $(Y_i, Z_i) \sim p_{h_\star}(\cdot, \cdot \mid X_i)$. For m independent examples, the likelihood is

$$\ell_m(h) = \prod_{i=1}^{m} p_h(Y_i, Z_i \mid X_i).$$

A maximum-likelihood learner can select h only if $\ell_m(h) \geq \ell_m(h_\star)$.

From consistency to likelihood. Recall the squared Hellinger distance from our likelihood lecture, now averaged over the input:

$$d_{\mathrm{H}}^2(h, h_\star) = \mathbb{E}_{X \sim \mathcal{D}}\left[\sum_{y,z}\left(\sqrt{p_h(y, z \mid X)} - \sqrt{p_{h_\star}(y, z \mid X)}\right)^2\right].$$

Its normalization gives the overlap identity

$$1 - \frac{d_{\mathrm{H}}^2(h, h_\star)}{2} = \mathbb{E}_{X \sim \mathcal{D}}\left[\sum_{y,z}\sqrt{p_h(y, z \mid X) p_{h_\star}(y, z \mid X)}\right].$$

The likelihood argument from that lecture bounds the chance that a fixed candidate fits at least as well as the teacher by this overlap raised to m; see also Foster, Block, and Misra [2, Proposition 2.1] and Doron-Arad, Mehalel, and Mossel [3, Lemma 14]. Defining

$$\mathcal{I}(h) = -\ln\left(1 - \frac{d_{\mathrm{H}}^2(h, h_\star)}{2}\right),$$

we obtain

$$\mathbb{P}[\ell_m(h) \geq \ell_m(h_\star)] \leq \left(1 - \frac{d_{\mathrm{H}}^2(h, h_\star)}{2}\right)^m = e^{-m\mathcal{I}(h)}.$$

This replaces the probability q_h^m of remaining consistent. The independence is across input problems; the steps within a CoT may depend on one another.

For deterministic candidates, the output distributions are point masses: their squared Hellinger distance is 0 where the outputs agree and 2 where they differ. Hence

$$d_{\mathrm{H}}^2(h, h_\star) = 2(1 - q_h), \qquad \mathcal{I}(h) = -\ln q_h.$$

Thus the deterministic information is a special case of the same likelihood comparison.

Why observing the CoT helps. With answers alone, the likelihood uses $p_h(y \mid x) = \sum_z p_h(y, z \mid x)$. For each x, y, Cauchy–Schwarz gives

$$\sum_z \sqrt{p_{h_\star}(y, z \mid x) p_h(y, z \mid x)} \leq \sqrt{p_{h_\star}(y \mid x) p_h(y \mid x)}.$$

Summing over y and averaging over x shows that hiding the CoT can only increase the overlap, and therefore can only decrease the information. In the noisy cutoff example, rejecting a student who lacks the prerequisite tells us nothing about the cutoff, but the observed score check still helps distinguish it.

The same inequality also explains why learning the complete output distribution is enough for answer prediction: its Hellinger distance bounds that of the answer marginal. The earlier likelihood lecture then converts Hellinger control into total variation control. Doron-Arad et al. [3] study the autoregressive version, where each step is sampled from one shared next-token predictor and the goal is to estimate the final-bit probability under squared loss. A full CoT supplies samples of intermediate decisions, not their underlying probabilities.

3 Learning efficiently

We return to deterministic, noiseless supervision. The statistical theorem allows an exhaustive search of $\mathcal{H}$. CoT can also make learning computationally efficient. To see the idea, extend the cutoff example to k unknown linear tests on $x \in \{0,1\}^n$:

$$z_j = \mathbf{1}\{w_j^\top x + b_j \geq 0\}, \quad j = 1, \ldots, k, \qquad y = z_1 \wedge \cdots \wedge z_k.$$

The weights and biases are unknown; the order of the tests and the final AND are fixed and known. The answer function is an *intersection of halfspaces*.

A rejection $y = 0$ says that some test failed, but not which one. For two tests, it could mean $(z_1, z_2) = (0,0)$, $(0,1)$, or $(1,0)$. Observing z supplies a label for each test. We can then learn the k thresholds separately, each by fitting linear inequalities using linear programming, and combine their predictions with AND. This gives a learner with polynomial running time and sample complexity.

Klivans and Sherstov [4, Theorem 1.1] prove that a polynomial-time answer-only learner for $k = n$ tests, valid for every target and input distribution, would give a polynomial-time algorithm for a particular version of the unique shortest vector problem. Assuming that this lattice problem has no randomized polynomial-time algorithm, answer-only learning is hard even with noiseless labels and even if the learner may return a predictor outside the target class. The hardness concerns a growing number of tests, not just two.

Joshi et al. prove polynomial-time CoT learning even when one threshold is reused on a moving window of bits [5, Corollary 4.3]. To rule out efficient answer-only learning, they assume that some class of constant-depth, polynomial-size threshold circuits cannot be learned in polynomial time from answers. They prove that an efficient answer-only learner for the repeated threshold would also learn those circuits, using a known encoding of the input, and would therefore contradict this assumption [5, Assumption 1 and Theorem 4.4]. Their CoT learning algorithm does not rely on the hardness assumption.

In the next lecture, we will consider a different source of guidance: generating several attempts and scoring them when a worked solution is not supplied.

References

[1] Awni Altabaa, Omar Montasser, and John Lafferty. CoT Information: Improved Sample Complexity under Chain-of-Thought Supervision. NeurIPS, 2025. https://arxiv.org/abs/2505.15927.

[2] Dylan J. Foster, Adam Block, and Dipendra Misra. Is Behavior Cloning All You Need? Understanding Horizon in Imitation Learning. NeurIPS, 2024. https://arxiv.org/abs/2407.15007.

[3] Ilan Doron-Arad, Idan Mehalel, and Elchanan Mossel. Stochastic Autoregressive Learning. arXiv:2608.07224, 2026. https://arxiv.org/abs/2608.07224.

[4] Adam R. Klivans and Alexander A. Sherstov. Cryptographic Hardness for Learning Intersections of Halfspaces. Journal of Computer and System Sciences, 75(1):2–12, 2009. https://www.cs.utexas.edu/~klivans/crypto-hs.pdf.

[5] Nirmit Joshi, Gal Vardi, Adam Block, Surbhi Goel, Zhiyuan Li, Theodor Misiakiewicz, and Nathan Srebro. A Theory of Learning with Autoregressive Chain of Thought. COLT, 2025. https://proceedings.mlr.press/v291/joshi25a.html.