

מודלי שפה גדולים ורשתות נוירונים

מהיסוד: שתי הרצאות לוח

מ-Stanford

Tokenization, מודל אוטו-רגרסיבי, SGD, Attention ו-ResNet: גזירה מתמטית על הלוח, בלי שקפים ובלי שיווק

מרצה ששמו לא נאמר בתמליל · Stanford Engineering (המקור המקורי; הועלה מחדש ב-X ע"י Komal Prajapati) · הועלה ל-X ב-2026-10-08, תאריך ההרצאה המקורית לא ידוע

משך: כ-155 דקות · X

סיכום אוטומטי — AI Briefing (aibriefing.dev)

על המסמך הזה

הסרטון הגיע מ-repost ב-X עם כיתוב שיווקי ("Skip Netflix tonight"), אבל בפועל זה צילום של שתי הרצאות לוח מקורס מבוא ל-Stanford Machine Learning ב-Stanford. בין שתיהן מופיע הבאמפר של Stanford Engineering. בחלק הראשון, עד בערך 1:16, המרצה בונה מודל שפה אוטו-רגרסיבי מאפס: Tokenization, פירוק ההסתברות בעזרת softmax, chain rule ו-temperature, ה-loss לאימון, ואז ה-Attention עם multi-head, masking, ומורכבות T^2 . החלק השני הוא הרצאה מוקדמת יותר בסילבוס, Deep Learning I: מסגרת הלמידה המפוקחת, cross-entropy, SGD, ReLU, שכבות, ResNet ו-LayerNorm.

בתמליל המרצה לא מזדהה בשמו. הוא מזכיר עמית בשם Chris שהעביר את ההרצאה על מודלים לינאריים, ומספר שהשתתף בפיתוח תאוריה על מינימומים מקומיים. הטענה המרכזית שלו היא שאין כאן קסם: LLM הוא רשת אחת שמנבאת את הטוקן הבא ומאומנת end-to-end על negative log-likelihood. כשנכנסים לפרטים, הרבה מהבחירות הארכיטקטוניות, כמו סוג האקטיבציה, מספר השכבות ב-residual block וסוג הנרמול, הן אמפיריות, והוא אומר את זה בעצמו בלי להתנצל.

איך לקרוא את המסמך

- 1. **תקציר מנהלים:** הטענות המרכזיות עם חותמות זמן, ומה לא נאמר.
- 2. **Tokenization:** למה subwords, גודל אוצר המילים, ומה הוא אמר על שינוי ה-tokenizer של Claude Code.
- 3. **המודל האוטו-רגרסיבי:** chain rule, embeddings, softmax, temperature, ו-loss האימון.
- 4. **Attention:** Q/K/V, causal mask, multi-head, residual, ונרמול, ועלות T^2 .
- 5. **מסגרת הלמידה ו-SGD (ההרצאה השנייה):** loss, cross-entropy, ולמה stochastic.
- 6. **רשתות נוירונים:** ReLU, שכבות, אקטיבציות, ResNet ו-LayerNorm.
- 7. מסקנות ושאלות פתוחות, ו-8. מילון מונחים.

הערות מקור המסמך נוצר אוטומטית מהתמליל של ההקלטה ומהלוחות שצולמו בה. התמליל הוא ASR, ולכן שמות ומונחים עלולים להשתבש (למשל QN במקום IDEM, Qwen במקום cloud code, Adam במקום Claude Code). הציטוטים הם מילה במילה מהתמליל ונבדקו מול חותמת הזמן שלהם. הכיתוב של ה-repost ב-X לא שימש מקור לעובדות, ושום דבר לא נוסף ממקורות חיצוניים.

1. תקציר מנהלים

זו לא הרצאה על ChatGPT או על Claude כמוצרים. זה שיעור אוניברסיטאי שגוזר על הלוח את המתמטיקה של pre-training: איך טקסט הופך לטוקנים, איך מנבאים את הטוקן הבא, ואיך עובד ה-Attention. אחריו מגיע שיעור יסודות על רשתות נוירונים ואופטימיזציה. הערך שלו בקוהרנטיות של הגזירה, לא בחידושים.

- **אלה שתי הרצאות, בסדר הפוך לסילבוס:** החלק על LLM מסתיים ב-[1:15:53], ומיד אחריו מתחילה הרצאה אחרת ("Chris has talked about a linear model"). החלק הראשון כבר מפנה ל-RMSNorm ול-residual שנלמדו בחלק השני.

- **Tokenization ברמת subword היא פשרה:** תווים בודדים דורשים יותר מדי צעדים ("You need so many tokens, so many steps." [1:31]), ומילים שלמות לא מאפשרות להכליל למילים נדירות. גודל אוצר המילים: "on the order of 10 to the 5, like 100K at least" [4:35].

- **שינוי ה-tokenizer של Claude Code הוא שמועה שלא אומתה:** המרצה מדבר על מעבר מ-1,000 ל-1,500 טוקנים לאותו טקסט, ומסייג בעצמו: "I didn't verify it myself but I read some news about this" [7:40].

- **ה-chain rule הוא מה שהופך את הבעיה לפתירה:** במקום התפלגות על $V|T$ רצפים מנבאים בכל צעד softmax על V אפשרויות בלבד [10:43].

- **Temperature מחדד את ההתפלגות או משטיח אותה:** חלוקת ה-logits ב- T קטן מרכזת כמעט את כל ההסתברות בטוקן הסביר ביותר, כלומר "You are taking the max as opposed to sampling from it." [32:15].

- **Attention הוא המקום היחיד שבו מידע עובר בין עמדות:** ה-MLP פועל על כל עמדה בנפרד עם פרמטרים משותפים [40:06]. ה-causal mask מוסיף מינוס אינסוף למשולש העליון [1:02:10], וה-heads ב-multi-head רצים במקביל ולא בשרשרת [1:08:14].

- **העלות הריבועית באורך הרצף היא צוואר הבקבוק:** $T^2 \cdot d_h$ פעולות [1:12:50]. Flash attention חוסך זיכרון, ושינויי ארכיטקטורה שמורידים את התלות ב- T עולים ביכולת הביטוי [1:15:53].

- **SGD הוא הכרח ולא רק אופטימיזציה:** המדרג הסטוכסטי הוא אומדן חסר הטיה של המדרג המלא, ו-"In 2026 now, I think it's kind of like, like one trillion tokens" [1:40:33]. ההשערה המקובלת לדבריו: כמעט אין מינימומים מקומיים רעים [1:43:37].

- **הרבה מהארכיטקטורה נקבע אמפירית:** GELU עדיף כי [2:11:16], "ReLU not smooth enough", residual block מסיבות לא מובנות [2:23:36], והנרמול [2:29:44] "mostly empirical".

מה לא נאמר

- **אי post-training:** לא מוזכרים RLHF, instruction tuning או שום דבר שהופך מודל בסיס לצ'אטבוט, אף שהכיתוב ב-X מבטיח הסבר על "how ChatGPT and Claude actually work".

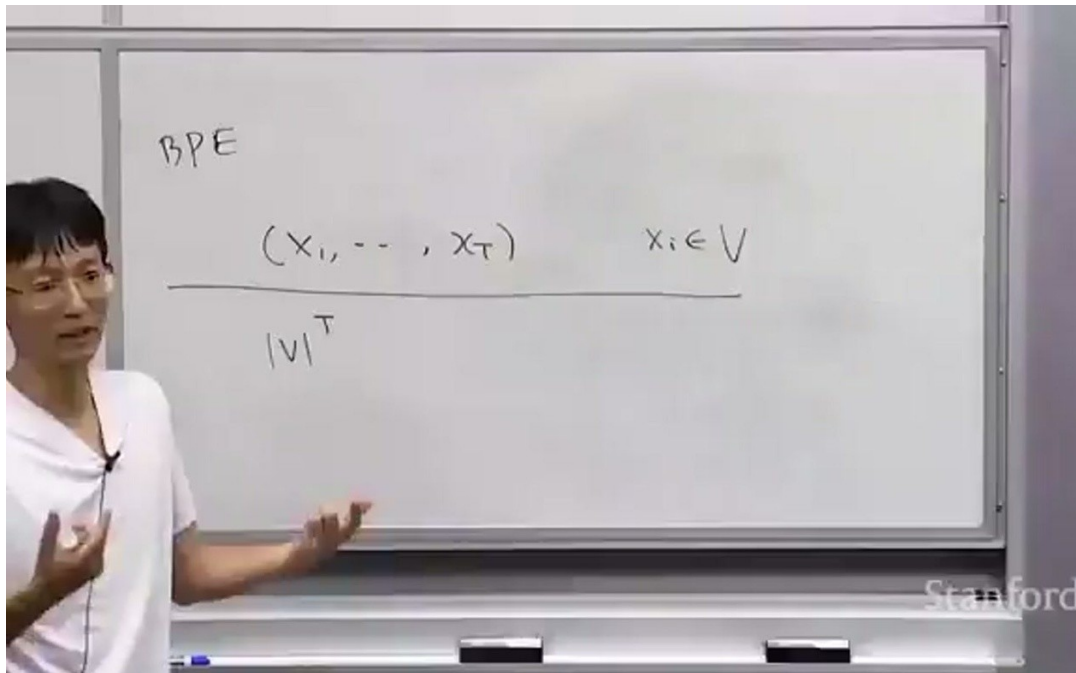
- **אי positional encoding:** ה-Attention כפי שהוצג על הלוח לא מקבל מידע על סדר הטוקנים, וההרצאה לא מסבירה מאיפה המודל יודע את המיקום.

- **אי KV cache, אי backpropagation בפירוט ואין את אלגוריתם BPE המלא:** הנגזרות מטופלות כ-auto differentiation שחור, ו-BPE מתואר רק כאלגוריתם greedy של מיזוגים תכופים.

- **מספרים שלא אומתו:** גודל אוצר המילים של Qwen 3.5 נשמע בתמליל "tokens 250", כנראה 250K (שיבוש או פליטת פה). הנתון של Claude Code מסויג על ידי המרצה עצמו.

- הארכיטקטורות שמחליפות את T^2 נדחו להרצאה של יום רביעי, שאינה בסרטון.

2. Tokenization: איך טקסט הופך למספרים

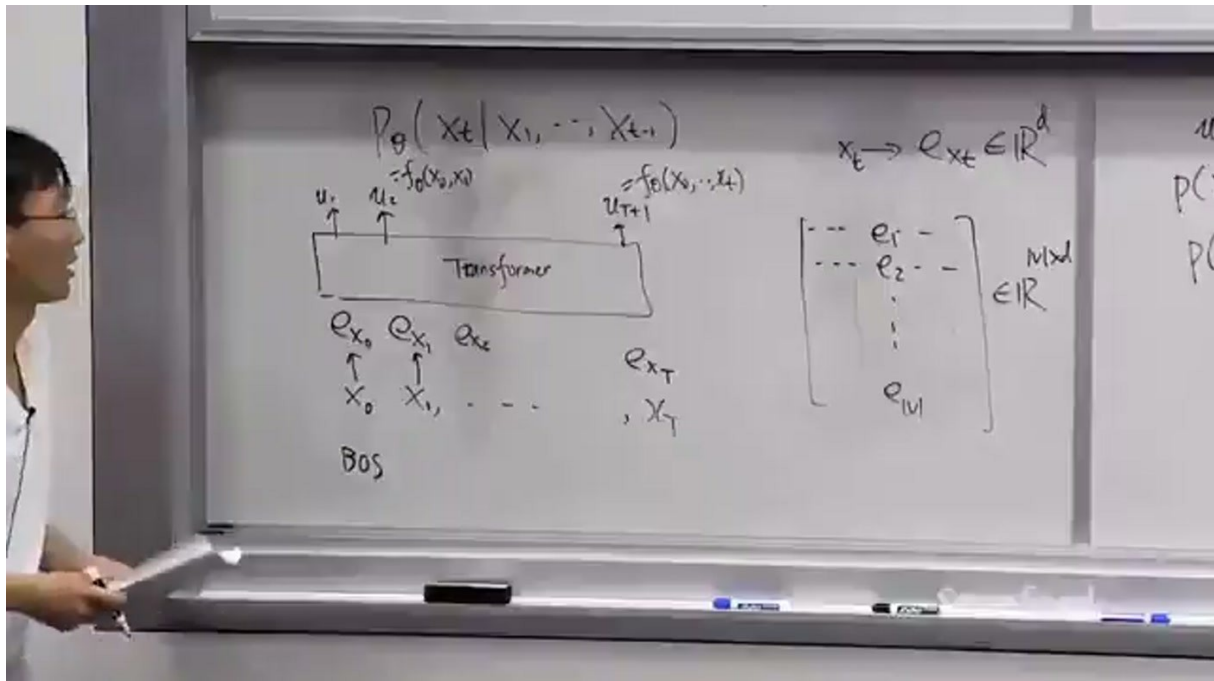


הלוח בפתיחה: BPE, רצף $x_1 \dots x_T$ שכל איבר בו שייך לאוצר המילים V , ומספר הרצפים האפשריים $|V|^T$

הטרנספורמר מקבל מספרים, ולכן צריך להחליט מהי יחידת הקלט הקטנה ביותר. ברמת תו בודד יוצאים רצפים ארוכים מדי. ברמת מילה, מילה נדירה כמו internationalization הופכת ליחידה נפרדת שלא חולקת ידע עם internationalized או עם nation. הפתרון הוא subwords. המרצה ממליץ לפתוח את ה-playground של OpenAI tokenizer ולראות, למשל, ש-"dog" עם הרווח שלפניו הוא טוקן אחד. אוצר המילים במודלים פתוחים "on the order of 10 to the 5, like 100K at least". הוא מזכיר את QN 3.5 (כנראה Qwen 3.5) עם "250", ומשמעות המספר לא ברורה מהתמליל. בכל אוצר מילים שומרים מקום לטוקנים מיוחדים, "maybe a thousand tokens", כמו "begin-of-sentence" [15:18].

לפי המרצה, אוצר המילים נבנה בעזרת BPE (byte-pair encoding): אלגוריתם greedy שממזג שוב ושוב את הצירופים התכופים ביותר. את הפרטים הוא מדלג עליהם כי "It's not that important". הנקודה המעשית שלו נוגעת לעלות: לכל חברה יש tokenizer קנייני, ולגבי Claude Code הוא אומר "So before it's 1,000 tokens. Now it's 1.5,000 tokens." [7:40]. כלומר אותו טקסט עולה יותר. הוא מציין במפורש שלא בדק את זה בעצמו.

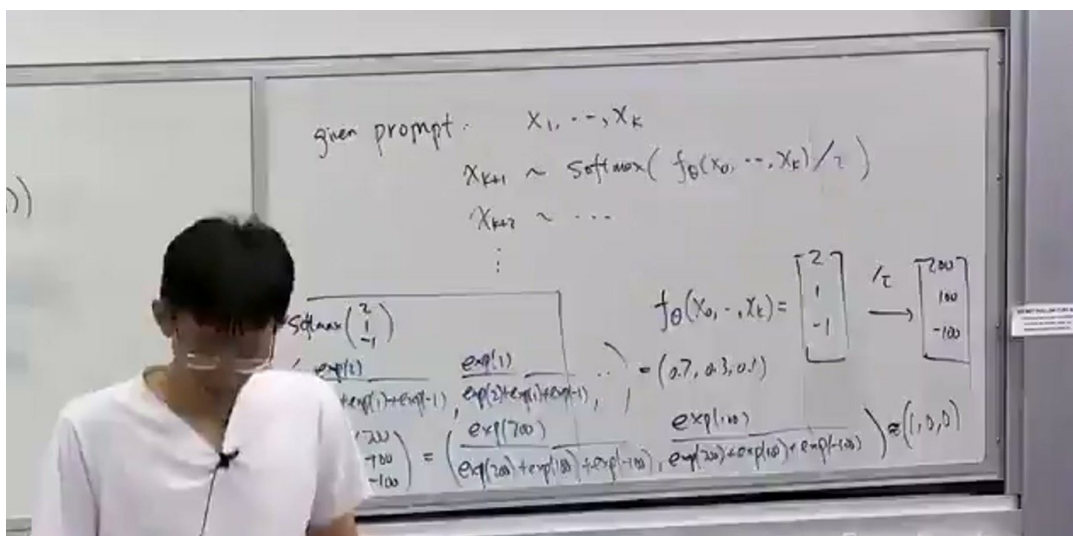
3. המודל האוטו-רגרסיבי: מ-chain rule ועד loss



הסכמה על הלוח: טוקנים (BOS) x_0 עד x_T עוברים דרך מטריצת embedding בגודל $|V| \times d$ אל הטרנספורמר, שמוציא בכל עמדה וקטור u_t

יש $|V|^T$ רצפים אפשריים, ולכן אי אפשר להצמיד הסתברות לכל רצף. מפרקים את ההתפלגות המשותפת למכפלה של התפלגויות מותנות, $p(x_1) \cdot p(x_2|x_1) \cdot \dots \cdot p(x_T|x_1..x_{T-1})$. כל גורם הוא softmax על $|V|$ אפשרויות בלבד [10:43], והיחס בין הקלט לפלט ממודל על ידי "a neural network, which is a transformer" [12:13]. מתרגמים כל ID לשורה במטריצת embedding, מוסיפים בהתחלה טוקן BOS קבוע, והפלט בעמדה t הוא וקטור logits באורך $|V|$ שעובר softmax. אחד הסטודנטים שאל על causality, והמרצה ענה שזה עדיין רק דרישה: "I'm insisting that" u_t is only a function of x_0 and x_2 minus 1 mask [23:04].

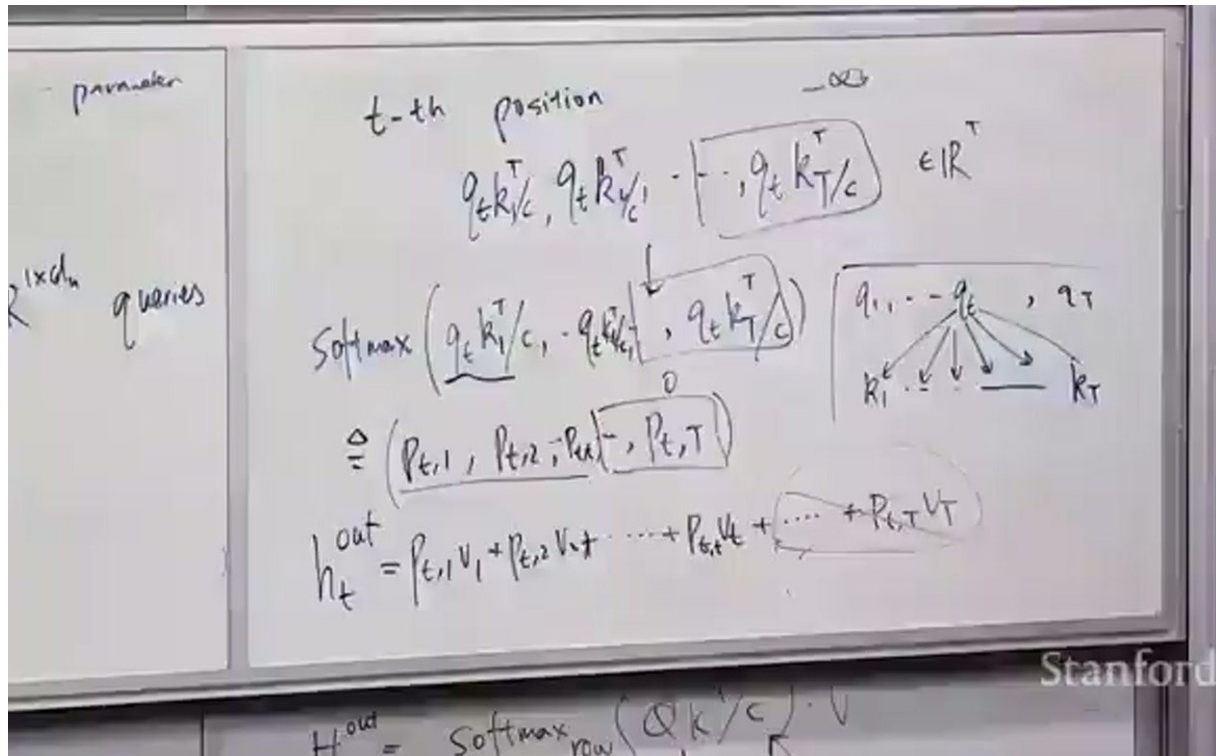
ביצירת טקסט (generation) דוגמים טוקן, מזינים אותו חזרה למודל וממשיכים. בדרך כלל מתחילים מ-prompt ולא מאפס [26:05]. Temperature מחלק את ה-logits ב-2. בדוגמה של המרצה, הווקטור (2, 1, -1) הופך עם τ קטן ל-(100, 200, -100), וה-softmax נותן בערך (0, 1, 0) [32:15]. האימון הוא מינימיזציה של סכום $\log(-\text{softmax}(f_\theta(x_0..x_{t-1})))$ באינדקס של הטוקן האמיתי, בעזרת optimizer כמו SGD או Adam (בתמליל IDEM) [36:59].



דוגמת ה-temperature על הלוח: הדגימה מ- $\text{softmax}(f_\theta/\tau)$, ומה קורה ל-logits ול-softmax כש- τ קטן

— המרצה [32:15] So basically what I'm trying to say is that if tau is a small number, so that you amplify the lodges, then you are sharpening the resulting distribution to make it more like a similar state distribution. And then as a result, you are just generating, you are just sampling from the most likely token. You are taking the max as opposed to sampling from it

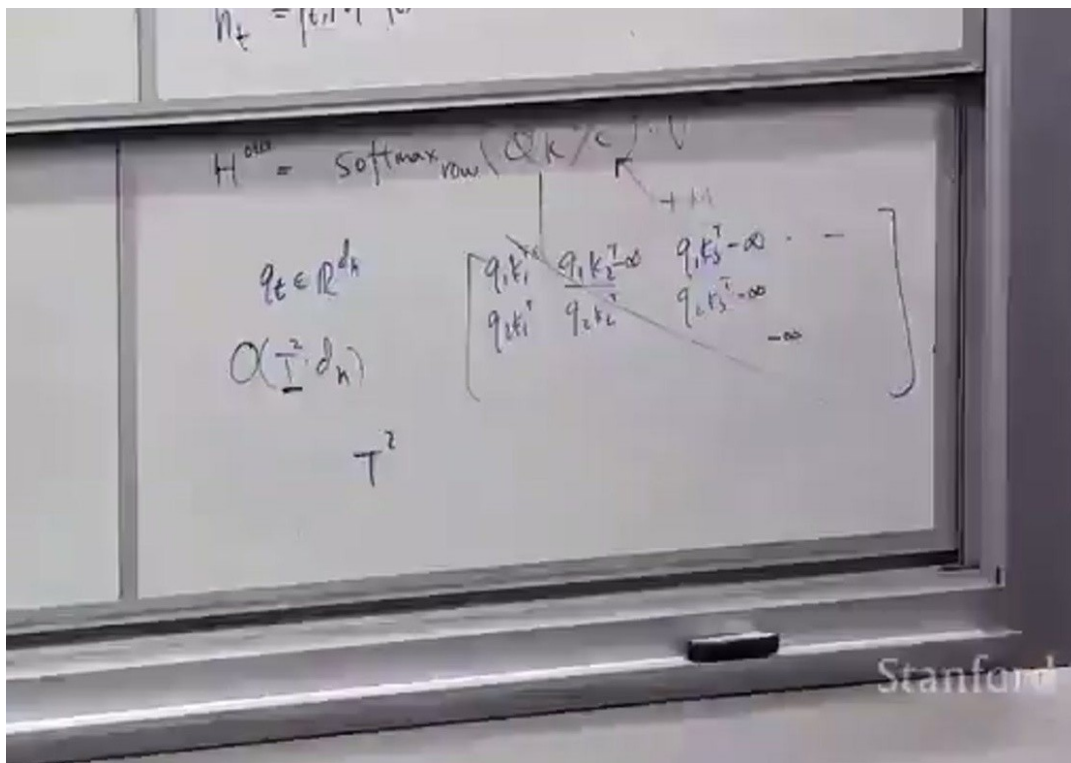
4. Attention: המנגנון שמחבר בין עמדות



Single-head attention בעמדה t : המכפלות $q_t^T k_j/c$, שנותנות את ההסתברויות $p_{t,1}..p_{t,T}$, והפלט h_t^out כצירוף משוקלל של ה-values

הטרנספורמר מחליף לסירוגין שכבות attention ושכבות MLP. ה-MLP חולק פרמטרים ופועל על כל עמדה בנפרד, ולכן "the only way that you are using, you are kind of like you fuse the input from different positions" [40:06] or different time steps is the attention". כל וקטור קלט מופצל ב- W_Q , ב- W_K וב- W_V (מ-d ל- d_h). ה-query של עמדה t נכפל בכל ה-keys, אחרי scaling עובר softmax, והפלט הוא ממוצע משוקלל של ה-values. המשקלות נלמדים, ולכן הארכיטקטורה היא רק "a template in some sense" [50:59]. למה לא MLP אחד ענק על כל הרצף? כי מספר הפרמטרים היה גדל עם אורך הרצף, שיכול להגיע למיליונים, ומשתנה אם אורך הרצף שונה בין אימון ל- $test$: "But that's too expensive." [54:17].

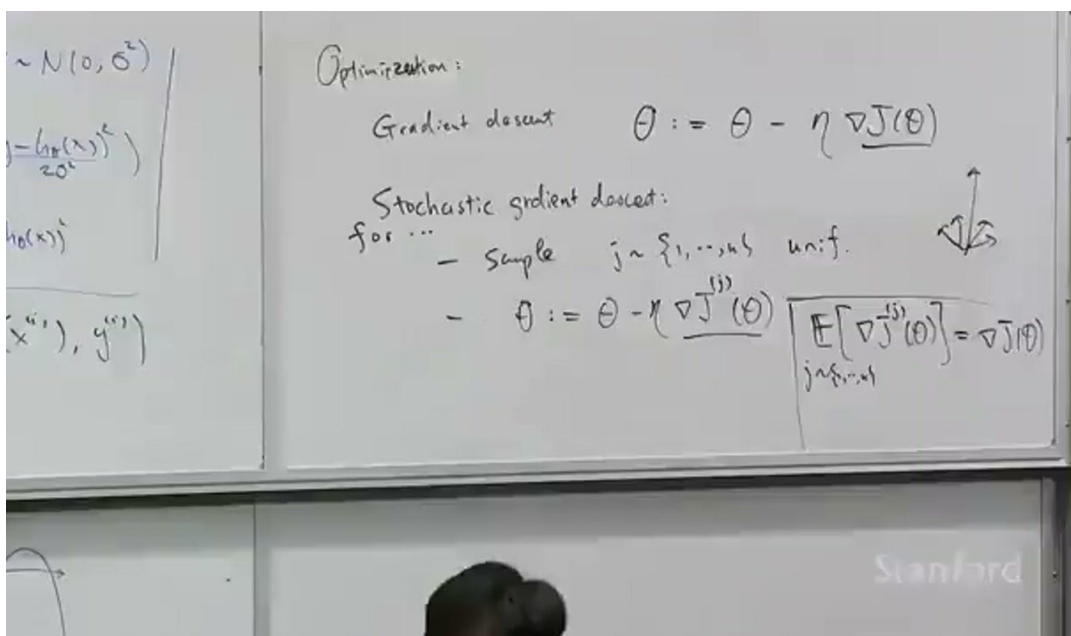
בכתיב מטריצוני, $H_{out} = \text{softmax_row}(QK^T/c) \cdot V$. בצורה הזו כל עמדה רואה גם את העתיד, ולכן מוסיפים מינוס אינסוף לכל המשולש העליון של מטריצת הציונים, וה-softmax מאפס אותם [1:02:10]. ב-multi-head יש כמה עותקים עם משקלות שונים. מחברים את הפלטים שלהם (concat) ומטילים אותם עם W^O . האינטואיציה שהוא מציע: head אחד מחפש ישויות, אחר סנטימנט. לשאלת סטודנט ענה: "No, no, they are all parallel. They are". [1:08:14] "parallel and then you aggregate." מסביב יש residual ונרמול (בדרך כלל RMSNorm), בגרסאות pre-norm או post-norm. הפרטים מופיעים ב-lecture notes.



d_h $H_{out} = \text{softmax}_{\text{row}}(QK^T/c) \cdot V$ מחשבות העלות: T^2 מכפלות פנימיות, כל אחת בממד d_h

— המרצה [1:15:53] So if you change the architecture, the dependency on t is better, but your architecture is less expressive. and then you may lose some performance to some degree

5. מסגרת הלמידה ו-SGD (ההרצאה השנייה)



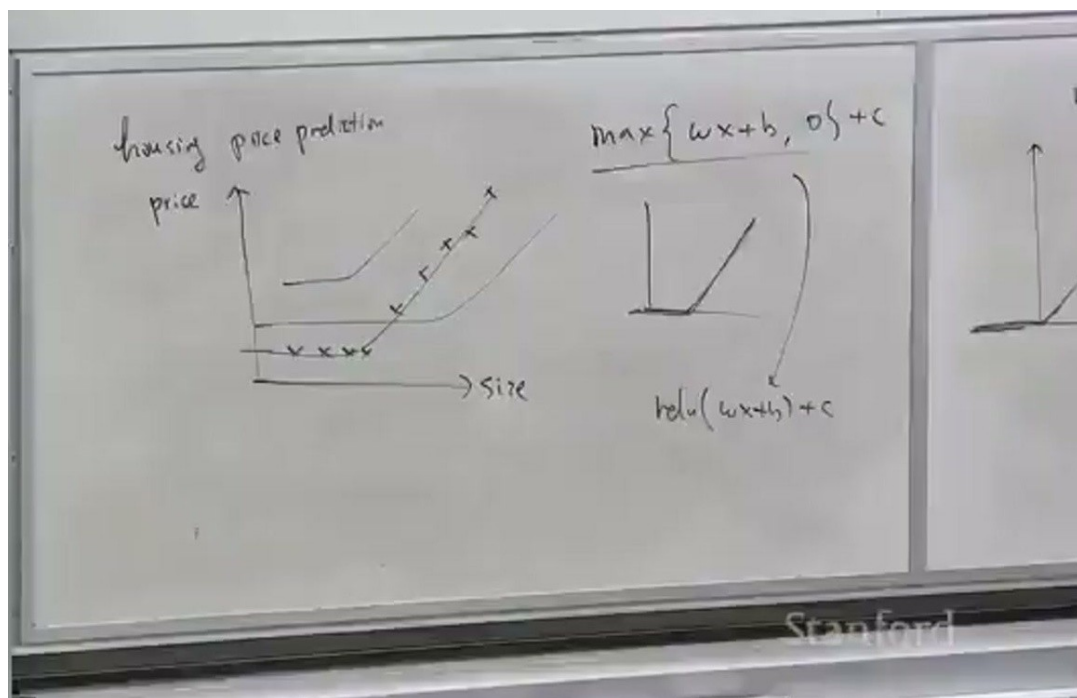
Gradient descent מול stochastic gradient descent: דגימה אחידה של j , עדכון לפי המדרג של דוגמה אחת, והתוחלת של המדרג הזה שווה למדרג המלא

ההרצאה השנייה, שעל הלוח נקראת Deep Learning I, מכילה את מה ש-Chris הציג למודלים לינאריים גם למודלים לא-לינאריים. ברגסיה ה-loss הוא mean squared error, ובסיווג רב-מחלקתי זה softmax על logits ואחריו negative log-likelihood, שהוא בדיוק cross-entropy ("cross-entropy is just called negative log-likelihood, which is just called cross entropy") loss ב-PyTorch [1:29:52]. לשאלת סטודנט הראה ש-MSE הוא NLL תחת הנחה של רעש גאوسی [1:32:54].

SGD דוגמים דוגמה אחת או mini-batch. המדרג שמתקבל שווה בתוחלת למדרג המלא. למה לא gradient descent מלא? כי ב-2015 היו כמיליון דוגמאות, ו-"In 2026 now, I think it's kind of like, like one trillion tokens, right? There's no chance you can compute all the gradient." [1:40:33] דוגמאות שונות מתואמים מאוד, וטעויות קטנות מתוקנות בהמשך. Mini-batch מאזן בין רעש לניצולת: עם 10,000 GPU ודוגמה אחת לצעד, "then most of the GPUs are actually idle." [1:48:11] על מינימומים מקומיים הוא מספר שהשתתף בפיתוח התאוריה, שלפיה בממדים גבוהים כמעט אין מינימומים רעים. הספירה נעשית בעזרת נוסחה מגאומטריה דיפרנציאלית שבתמליל נשמעת "cataracts formula", כנראה Kac-Rice, אבל זה לא ודאי.

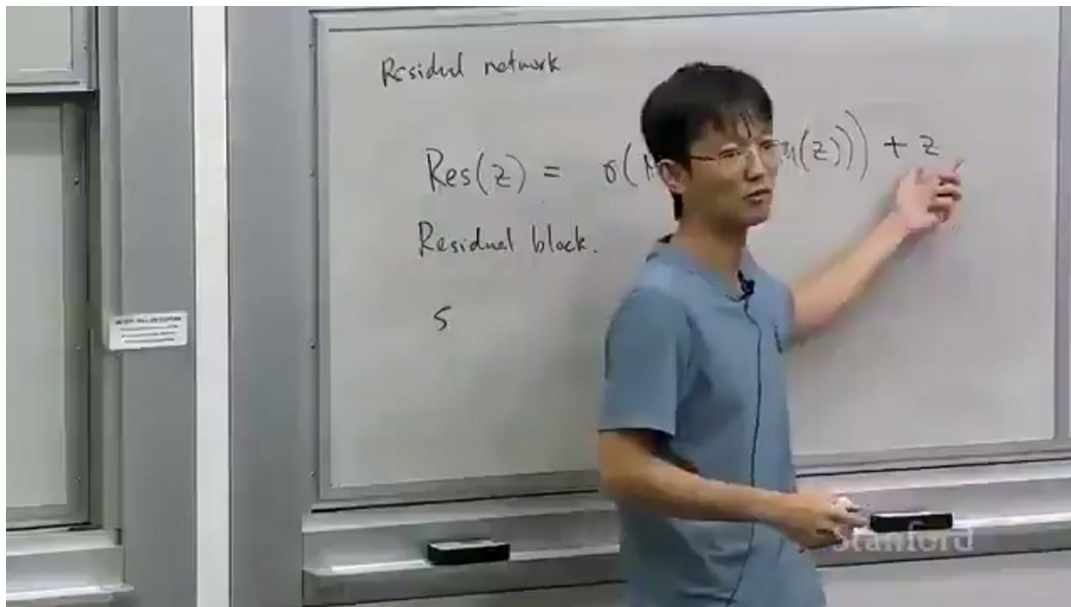
— **המרצה [1:43:37]** Maybe there's even no bad local minimum. All the local minimum are actually global minimum. It's kind of like if you think about it, you are skiing in a mountain

6. רשתות נוירונים: ReLU, שכבות, ResNet ונרמול



דוגמת מחירי הדירות: עקומה שטוחה ואחריה לינארית, שמבוטאת כ- $\max\{wx+b, 0\}+c$, כלומר $\text{relu}(wx+b)+c$

המוטיבציה היא דוגמת מחיר דירה מול גודלה: מודל לינארי לא מתאר עקומה שמתחילה שטוחה, ו- $\text{ReLU}(wx+b)$ כן. השם activation מגיע מהביולוגיה, שבה נוירון יורה רק מעל סף. בתחילת ה-deep learning השתמשו ב-sigmoid מאותה סיבה. אחר כך הוא מוסיף תכונות ביניים כמו walkability ו-school quality כפונקציות לא-לינאריות של הקלט, ומגיע לשכבה בכתוב מטריוני: $a = \text{ReLU}(Wx+b)$, עם W בממד $d \times m$ ו- m פרמטרים. רשת עמוקה היא הרכבה של מכפלות מטריצה ואקטיבציות, והממדים חייבים להתאים משכבה לשכבה. בין האקטיבציות הוא מציג So ReLU is actually not necessarily great" (בתמליל Jailu), ומודה: "because it's not smooth enough. But other than that, I don't really know exactly." [2:11:16]



Residual block על הלוח: $Res(z) = \sigma(MM(\sigma(MM(z)))) + z$, ורשת residual כשרשרת של בלוקים כאלה

ResNet מוצג כסיפור ולא כהוכחה. אם ייצוג הביניים z כבר קרוב ל- y , עדיף למדל רק את ההפרש, ומשם מגיעים ל- $y = f(z) + z$. את ההפרש ממדלים בשתי שכבות ולא באחת [2:22:04]. LayerNorm מנרמל לפי ממוצע וסטיית תקן אמפיריים, ואז משאיר פרמטרים נלמדים γ ו- β כדי שהאימון יחליט לאיזו סקאלה לנרמל. RMSNorm מוותר על חיסור הממוצע ומשאיר רק את γ . גם כאן המרצה לא מתיימר להסביר: "I think it's mostly empirical." [2:29:44]

— המרצה [2:23:36] You can do two layers, three layers, L layers, and eventually I think two or three layers are the best. Also for reasons we don't fundamentally understand, necessarily

7. מסקנות ושאלות פתוחות

- **למדוד טוקנים לכל מודל בנפרד:** מספר הטוקנים לאותו טקסט משתנה בין tokenizers ובין גרסאות, ושינוי tokenizer משנה את החשבון גם בלי שינוי במחיר לטוקן [7:40]. הכלי שהוא מציע הוא ה-playground של OpenAI tokenizer [4:35].
- **Temperature היא כוונון של חדות ההתפלגות:** τ קטן מתקרב ל- $\arg\max$ ו- τ גדול משטחי. זו ההבנה שצריך כשבוחרים פרמטרים למשימות דטרמיניסטיות מול משימות יצירתיות [32:15].
- **להתייחס לאורך ההקשר כעלות ריבועית:** $T^2 \cdot d_h$ פעולות וזיכרון. Flash attention פותר את הזיכרון, וארכיטקטורות חלופיות משלמות ביכולת ביטוי [1:12:50, 1:15:53].
- **להשלים ממקורות אחרים את מה שלא כוסה:** positional encoding ו-KV cache ו-post-training. בלעדיהם התמונה של "איך ChatGPT עובד" חלקית.
- **לקרוא את הפרטים ב-lecture notes של הקורס:** pre-norm מול post-norm, האקטיבציות שבשימוש בפועל ואלגוריתם BPE. המרצה מפנה אליהם במפורש.

שאלה פתוחה	למה היא פתוחה	היכן בהרצאה
האם ה-tokenizer של Claude Code באמת מייצר פי 1.5 טוקנים?	המרצה מסתמך על חדשות שלא בדק	[7:40]
מה גודל אוצר המילים של Qwen 3.5?	בתמליל נשמע "250 tokens", כנראה 250K	[4:35]
למה אקטיבציות חלקות כמו GELU עובדות טוב יותר?	"I don't really know exactly", בעיקר ניסוי וטעייה	[2:11:16]
למה שתיים-שלוש שכבות ב-residual block הן האופטימום?	"reasons we don't fundamentally understand"	[2:23:36]
האם באמת אין מינימומים מקומיים רעים?	השערה מקובלת, מוכחת רק במקרים מסוימים	[1:43:37]

שאלה פתוחה	למה היא פתוחה	היכן בהרצאה
אילו ארכיטקטורות מורידות את T^2 ובאיזה מחיר?	נדחה להרצאת יום רביעי, שאינה בסרטון	[1:15:53]
מאיפה ה-Attention יודע את סדר הטוקנים?	לא נאמר בהרצאה	—

8. מילון מונחים

מונח	משמעות
Autoregressive	מודל שמייצר טוקן אחרי טוקן, וכל טוקן מותנה בכל מה שלפניו
Tokenization / BPE	חלוקת טקסט ליחידות BPE. subword. בונה את אוצר המילים במיזוג greedy של צירופים תכופים
Vocabulary (V)	קבוצת הטוקנים האפשריים, בדרך כלל 100K ומעלה, כולל טוקנים מיוחדים שמורים
Embedding	שורה במטריצה בגודל $V \times d$ שמתרגמת ID של טוקן לוקטור
BOS token	טוקן קבוע בתחילת הרצף שממנו מתחילה היצירה
Logits / Softmax	ציונים גולמיים על כל אוצר המילים, וההמרה שלהם להתפלגות הסתברות
Temperature (τ)	מחלק את ה- τ logits. קטן מחדד לכיוון argmax ו- τ גדול משטיח
Negative log-likelihood / Cross-entropy	ה- loss של האימון: מינוס לוג ההסתברות של הטוקן או המחלקה האמיתיים
Attention (Q/K/V)	מנגנון שבו כל עמדה משקללת values של עמדות אחרות לפי דמיון query-key
Causal masking	הוספת מינוס אינסוף למשולש העליון כדי שעמדה לא תראה את העתיד
Multi-head attention	כמה attention heads במקביל עם משקלות שונים, שהפליטים שלהם מחוברים ומוטלים עם W^O
MLP	רשת קטנה שמופעלת על כל עמדה בנפרד עם פרמטרים משותפים
Flash attention	חישוב attention בבולוקים שחוסך את שמירת מטריצת $T \times T$ בזיכרון
SGD / mini-batch	עדכון לפי מדרג של דוגמה אחת או של קבוצה קטנה. בתוחלת זה המדרג המלא
ReLU / GELU	פונקציות אקטיבציה: $\max(0, t)$, ו-GELU שהיא גרסה חלקה עם שקע קטן מתחת לאפס
Residual connection (ResNet)	חיבור הקלט לפלט הבלוק, כך שהבלוק ממדל רק את ההפרש
LayerNorm / RMSNorm	נרמול לפי ממוצע וסטיית תקן (או RMS בלבד) עם פרמטרי סקאלה נלמדים