Structured Writing for Professionals

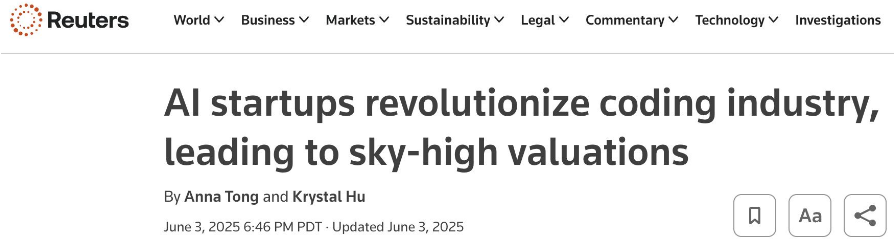
English 170
Stanford University, Fall 2025
Mihail Eric

The Modern Software Developer

CS146S
Stanford University, Fall 2025
Mihail Eric

Introduction and How LLMs are Made

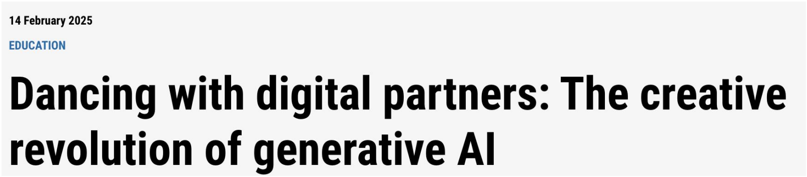
State of the World: 2025



AI startups revolutionize coding industry, leading to sky-high valuations

By Anna Tong and Krystal Hu

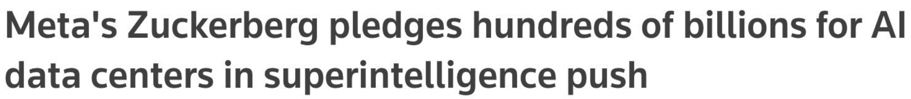
June 3, 2025 6:46 PM PDT · Updated June 3, 2025



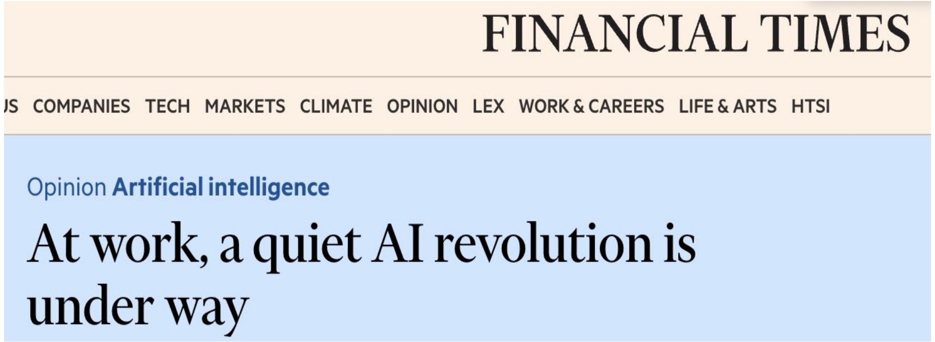
14 February 2025

EDUCATION

Dancing with digital partners: The creative revolution of generative AI



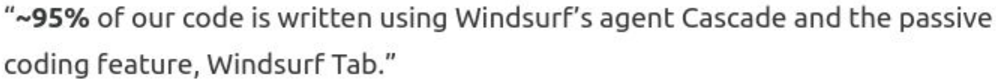
Meta's Zuckerberg pledges hundreds of billions for AI data centers in superintelligence push



Opinion Artificial intelligence

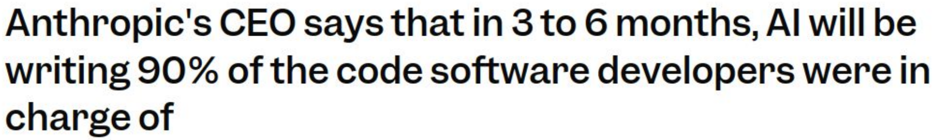
At work, a quiet AI revolution is under way

Bad News

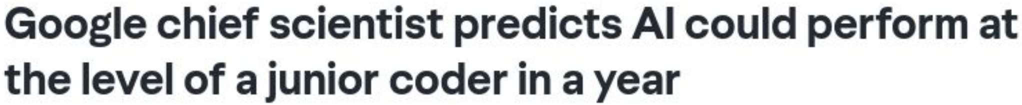


"~95% of our code is written using Windsurf's agent Cascade and the passive coding feature, Windsurf Tab."

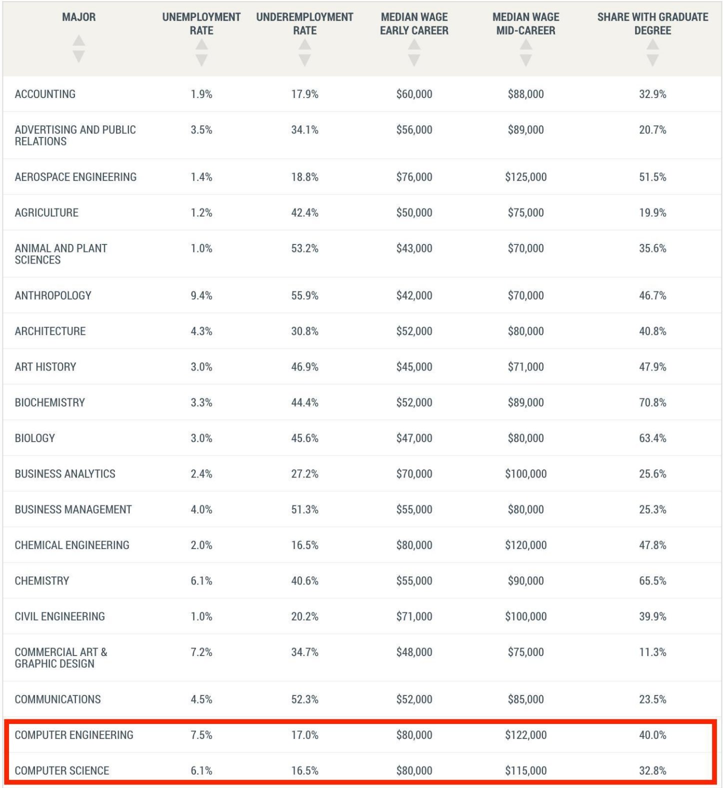
- Windsurf team



Anthropic's CEO says that in 3 to 6 months, AI will be writing 90% of the code software developers were in charge of



Google chief scientist predicts AI could perform at the level of a junior coder in a year



MAJOR	UNEMPLOYMENT RATE	UNDEREMPLOYMENT RATE	MEDIAN WAGE EARLY CAREER	MEDIAN WAGE MID-CAREER	SHARE WITH GRADUATE DEGREE
ACCOUNTING	1.9%	17.9%	$60,000	$88,000	32.9%
ADVERTISING AND PUBLIC RELATIONS	3.5%	34.1%	$56,000	$89,000	20.7%
AEROSPACE ENGINEERING	1.4%	18.8%	$76,000	$125,000	51.5%
AGRICULTURE	1.2%	42.4%	$50,000	$75,000	19.9%
ANIMAL AND PLANT SCIENCES	1.0%	53.2%	$43,000	$70,000	35.6%
ANTHROPOLOGY	9.4%	55.9%	$42,000	$70,000	46.7%
ARCHITECTURE	4.3%	30.8%	$52,000	$80,000	40.8%
ART HISTORY	3.0%	46.9%	$45,000	$71,000	47.9%
BIOCHEMISTRY	3.3%	44.4%	$52,000	$89,000	70.8%
BIOLOGY	3.0%	45.6%	$47,000	$80,000	63.4%
BUSINESS ANALYTICS	2.4%	27.2%	$70,000	$100,000	25.6%
BUSINESS MANAGEMENT	4.0%	51.3%	$55,000	$80,000	25.3%
CHEMICAL ENGINEERING	2.0%	16.5%	$80,000	$120,000	47.8%
CHEMISTRY	6.1%	40.6%	$55,000	$90,000	65.5%
CIVIL ENGINEERING	1.0%	20.2%	$71,000	$100,000	39.9%
COMMERCIAL ART & GRAPHIC DESIGN	7.2%	34.7%	$48,000	$75,000	11.3%
COMMUNICATIONS	4.5%	52.3%	$52,000	$85,000	23.5%
COMPUTER ENGINEERING	7.5%	17.0%	$80,000	$122,000	40.0%
COMPUTER SCIENCE	6.1%	16.5%	$80,000	$115,000	32.8%

Good News

- Software developers have the potential to be more productive than they have ever been in history
- With AI coding an engineer can pick up tech stacks and tools at an unprecedented pace
- You won’t be replaced by AI. You’ll be replaced by a competent engineer who knows how to use AI.

The Modern Software Developer

This is **not** the “vibe coding” class

10 weeks in 2 slides

The Takeaway

- Human-agent engineering
  - Focus on the skills that are not yet replaced by AI systems
  - Business understanding
  - Become the tech lead
- LLMs are only as good as you are
  - Good context leads to good code
  - If you can’t understand your codebase, neither will an LLM

The Takeaway

- Read and review **a lot** of code
    - Learn to discern good from bad, wrong software
    - Have good taste
- Experiment aggressively
    - There are no established software patterns yet
    - Everyone is still figuring it out
    - This class will introduce many workflows and tools - figure out what works for you

Course Logistics

- A bit about me
  - Stanford undergrad/grad
  - Head of AI at a stealth startup in the sales space
  - Built first LLMs at Amazon Alexa
  - Founded and sold an ML education startup
  - Founded a YC-backed AI coding company
- 1 awesome CA
  - Febie Lin

Course Logistics

- https://themodernsoftware.dev
- Lectures
  - Mon/Fri 8:30-9:20 am
- Deliverables
  - 9 assignments (1x/week) focusing on lecture material practice
    - https://github.com/mihail911/modern-software-dev-assignments
  - 1 final open-ended project in which you will exercise AI coding principles we cover
- Grading
  - 80/15/5 breakdown for project/assignments/participation
- Something pretty awesome
  - Guest lectures from founders leading top AI developer startups today
  - $100s of millions raised, billions in valuation
  - Don’t miss these talks!

How LLMs Work in 5 Slides
(For Engineers)

Basics

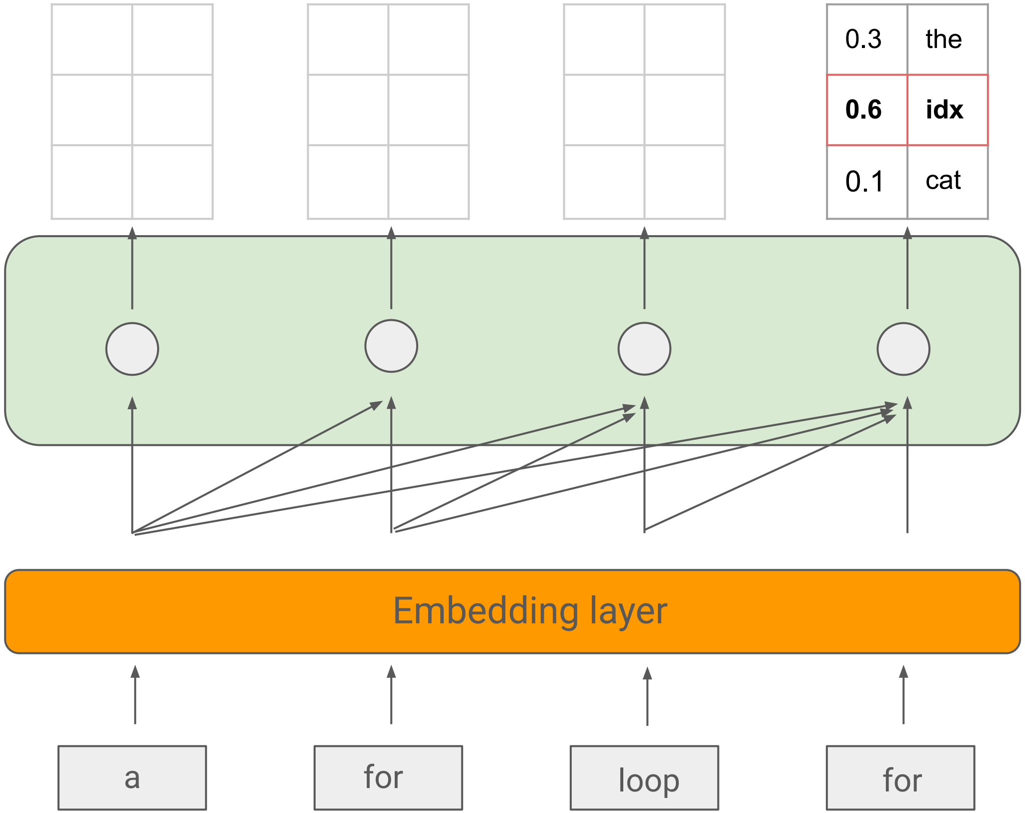
- LLMs (large language models) are autoregressive models for next-token prediction

$$P(x_1, x_2, \ldots, x_T) = \prod_{t=1}^{T} P(x_t | x_1, x_2, \ldots, x_{t-1})$$

$$\Big\downarrow$$

$$P(\mathrm{write, a, for, loop, for, idx, in, range\ldots}) = P(\mathrm{range}|\mathrm{write, a, for, loop, for, idx, in}) \cdot P(\mathrm{in}|\mathrm{write, a, for, loop, for, idx}) \cdot \ \ldots \ \cdot P(\mathrm{a}|\mathrm{write}) \cdot P(\mathrm{write})$$

Basics



0.3	the
0.6	**idx**
0.1	cat

Get probability distribution over most likely next token

Transformers layers (12-96+) using self-attention mechanism (Viswani et. al. 2017)

Convert tokens into fixed-dimensional numerical vectors (~1-3K dimensions)

Tokenize inputs using fixed vocabulary

Training Process

- Stage 1
    - Self-supervised pretraining
    - Teach the model notion of language on a variety of often public data sources
    - 100s of billions to trillion+ tokens (language and code)
    - Common Crawl, Wikipedia, StackExchange, Public Github repos
    - *Write a for loop* → **that could be used in a piece of code**
- Stage 2
    - Supervised finetuning
    - Teach model to follow instructions
    - High-quality, curated prompt-response pairs ("*what is the capital of Croatia*" -> "*Zagreb is the capital*")
    - Tens of thousands to 100s of thousands of pairs
    - *Write a for loop* → **ok here's a for loop…**
- Stage 3
    - Preferencing tuning
    - Align model outputs with human preferences (helpfulness, correctness, readability)
    - Collect pairs of outputs for same prompt and train reward model to predict preferred output
    - Tens of thousands to 100s of thousands of human-labeled comparisons
    - *Write a for loop* → **for idx in range(10):**

Training Process

- Reasoning models
  - Extend training with chain-of-thought reasoning traces
  - Tool-use integration
  - Get human preferences on reasoning steps
  - Reinforcement learning to learn how to evaluate reasoning traces, backtrack, etc
- Size
  - GPT-3/Claude 3.5 Sonnet - 175B parameters
  - LLaMA 3.1 - 405B parameters
  - GPT-4 - 1.8T (reported)

In practice

- Strengths
  - Expert-level code completion
  - Code understanding
  - Code fixing
- Limitations
  - Hallucinations
    - Generating non-existent/out-of-date APIs (mitigated with robust context engineering)
  - Context window limits
    - ~100-200K tokens but not all are created equal
  - Latency
    - Seconds to minutes per request depending on task (plan and delegate accordingly)
  - Cost
    - $1-3 per million input tokens, $10+ per million output tokens for best models

Questions?