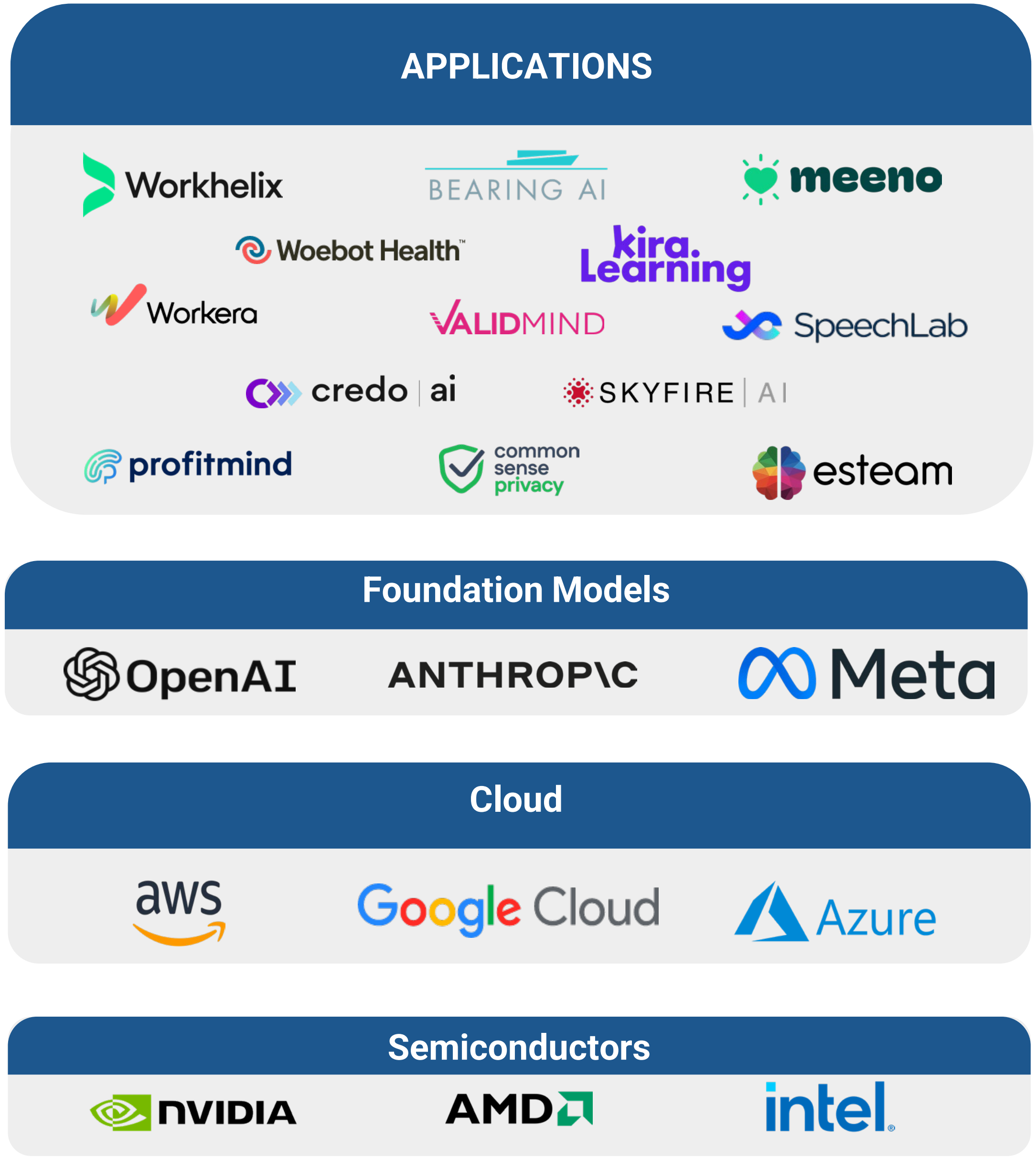
# AI Trends

Andrew Ng

# The AI Stack: Where are the biggest opportunities?



APPLICATIONS

Workhelix
BEARING AI
meeno
Woebot Health
kira Learning
Workera
VALIDMIND
SpeechLab
credo ai
SKYFIRE AI
profitmind
common sense privacy
esteam

Foundation Models

OpenAI
ANTHROPIC
Meta

Cloud

aws
Google Cloud
Azure

Semiconductors

NVIDIA
AMD
intel

Even though a lot of attention is on AI technology (esp. foundation models) most of the opportunities will be in building AI applications.

This is driving significant demand for inference workloads (more, faster tokens).

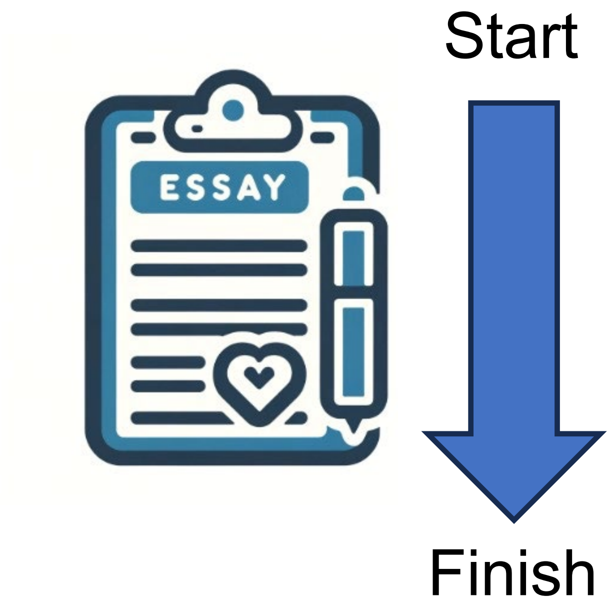
Andrew Ng

# Agentic AI workflows

# Agentic AI

## Non-agentic workflow (zero-shot):

Please type out an essay on topic X from start to finish in one go, without using backspace.



Start

Finish

## Agentic workflow:

Write an essay outline on topic X

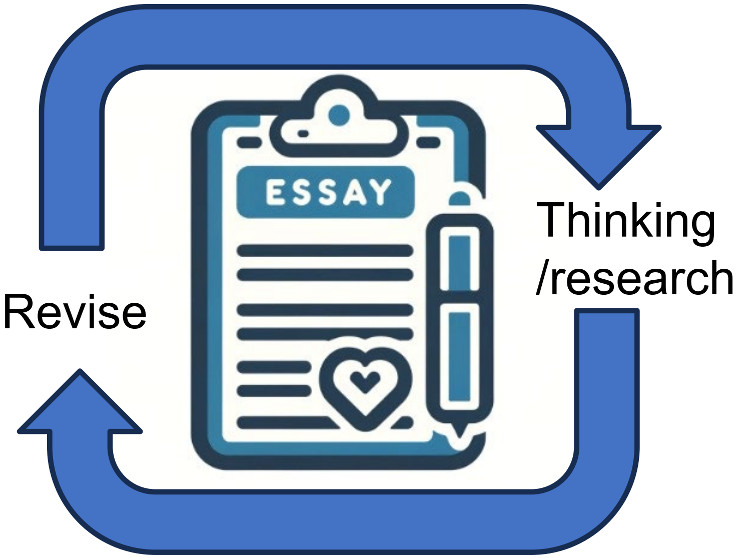
Do you need any web research?

Write a first draft.

Consider what parts need revision or more research.

Revise your draft.

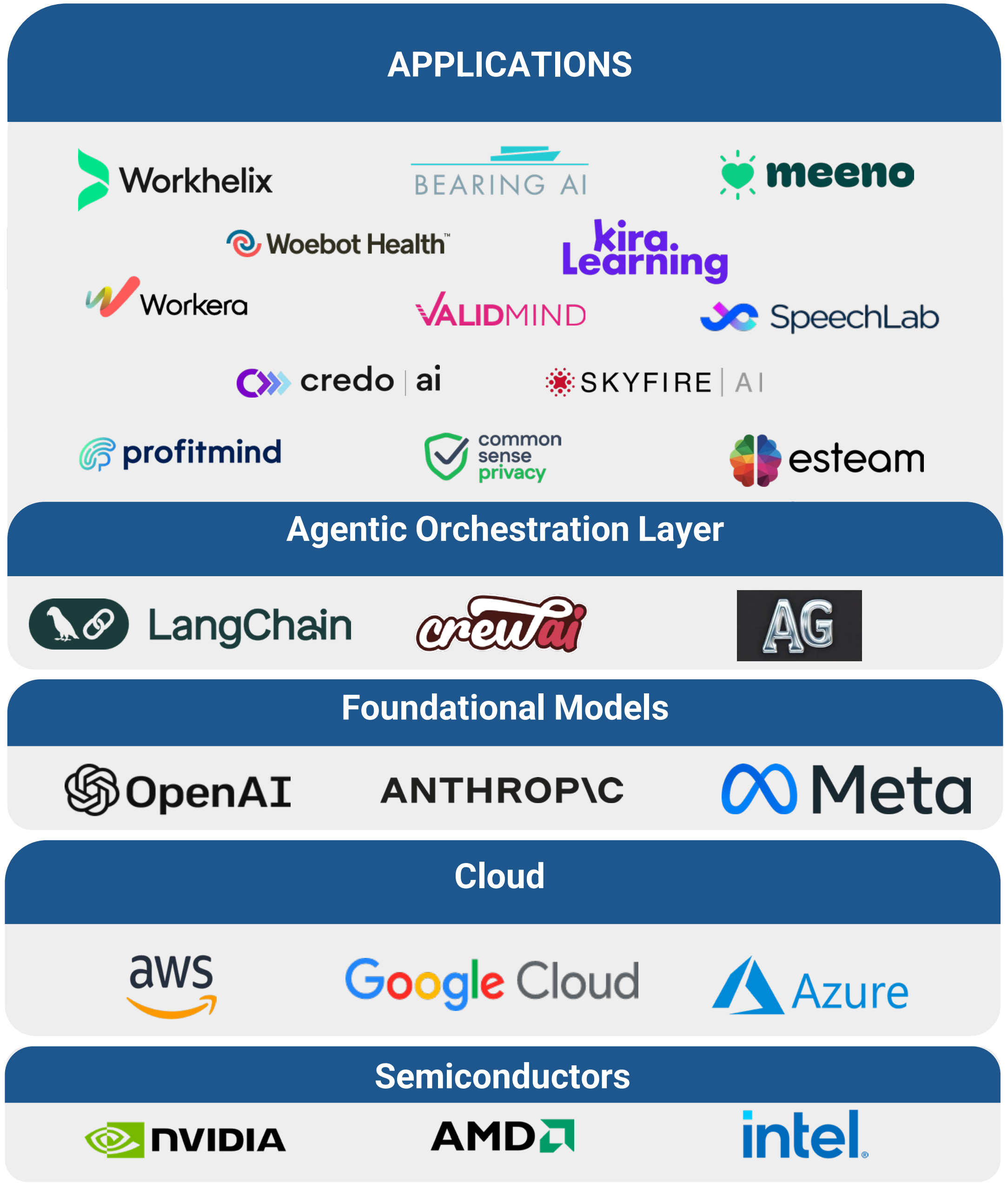
….



Revise

Thinking /research

# The AI Stack: Where are the biggest opportunities?



APPLICATIONS

Workhelix, BEARING AI, meeno, Woebot Health, kira Learning, Workera, VALIDMIND, SpeechLab, credo ai, SKYFIRE AI, profitmind, common sense privacy, esteam

Agentic Orchestration Layer

LangChain, crewai, AG

Foundational Models

OpenAI, ANTHROPIC, Meta

Cloud

aws, Google Cloud, Azure

Semiconductors

NVIDIA, AMD, intel

Even though a lot of attention is on AI technology (esp. foundation models) most of the opportunities will be in building AI applications.

New agentic orchestration layer

Andrew Ng

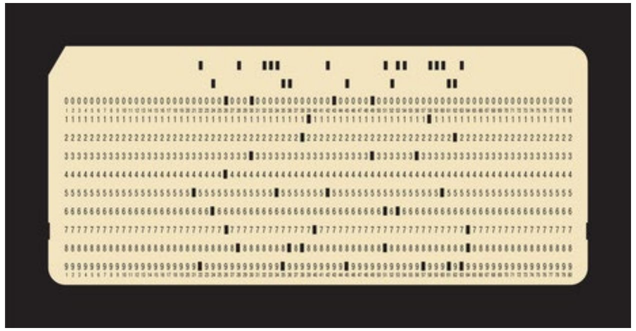
# Important AI trends

# Five important technical AI trends

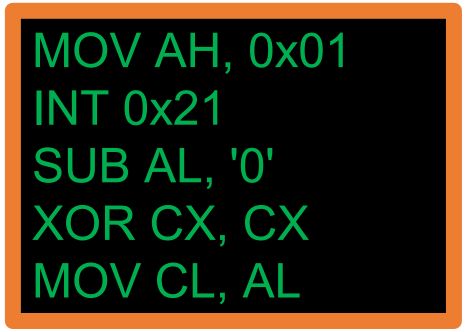
1. AI coding assistance
2. Fast prototyping
3. Visual AI
4. Data engineering
5. Talent gap

# 1. AI coding assistance – making coding pervasive

Some are advising people not to learn coding on the grounds AI will automate it. This is bad career advice.



Punch cards
→ Keyboard



```
MOV AH, 0x01
INT 0x21
SUB AL, '0'
XOR CX, CX
MOV CL, AL
```

Assembly
→ Modern languages

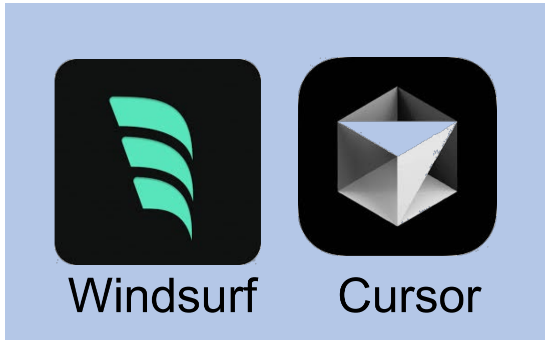
```
               VIM - Vi Improved

               version 7.4.629
            by Bram Moolenaar et al.
          Modified by
    Vim is open source and freely distributable

            Help poor children in Uganda!
   type  :help iccf              for information

   type  :q                      to exit
   type  :help          or       for on-line help
   type  :help version7          for version info
```

Text editor
→ IDE



AI coding
assistance

As coding becomes easier, people should code a lot more!

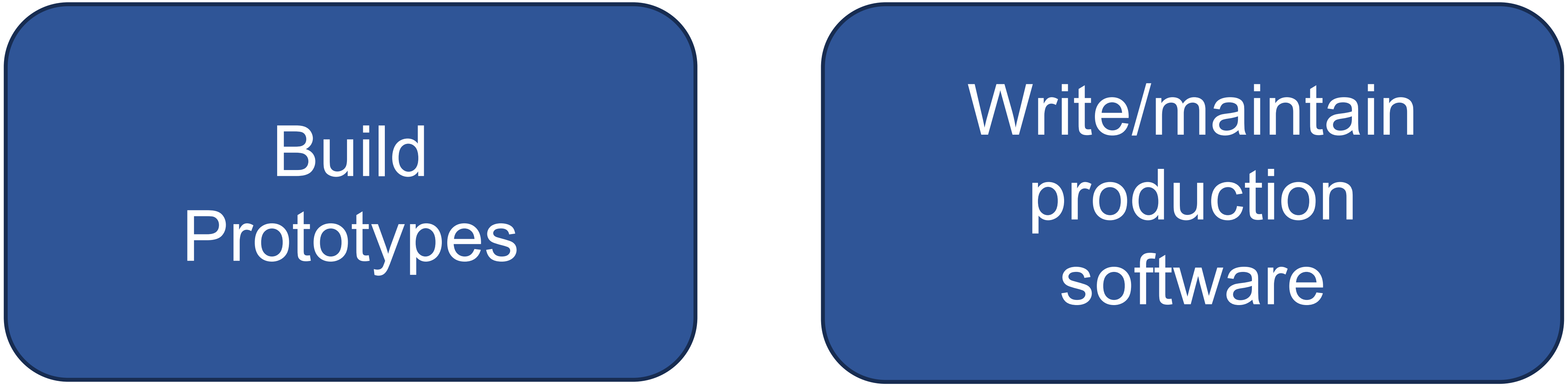
For professional engineers, this is already significantly boosting efficiency.

Andrew Ng

# Getting computers to do what you want

From: Generative AI for Everyone

# 2. Fast prototyping



Build Prototypes

Write/maintain production software

10x faster!

30-50% faster?

- Standalone prototypes/applications require less integration with legacy data sources and infrastructure
- Lower requirements on reliability, scalability, or even security, if prototyping to test basic functionality

- To pursue innovation, you can build 20 prototypes to see what works

# 2. Fast prototyping

Build Prototypes

10x faster!

Write/maintain production software

30-50% faster?

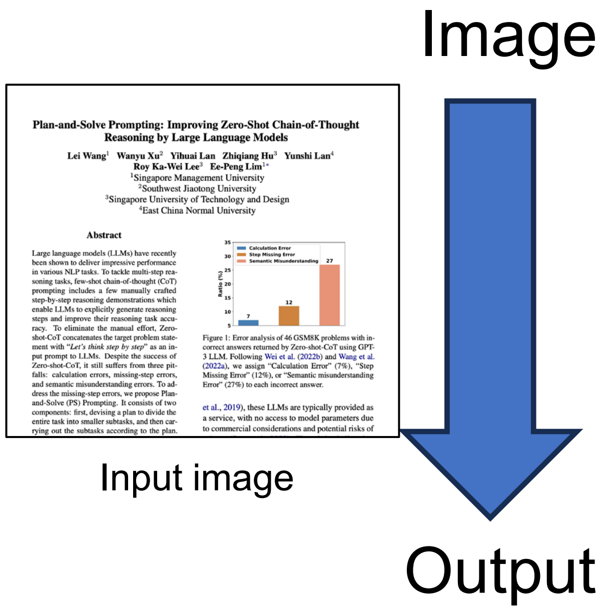
~~Move fast and break things.~~

Move fast and be responsible.

# 3. Visual AI

- The text processing revolution has arrived. The image processing revolution is coming, and will enable many new visual applications in manufacturing, self-driving, etc.

**Agentic Document Extraction**



**Plan-and-Solve Prompting: Improving Zero-Shot Chain-of-Thought Reasoning by Large Language Models**

**Lei Wang**[1] **Wanyu Xu**[2] **Yihuai Lan** **Zhiqiang Hu**[3] **Yunshi Lan**[4]
**Roy Ka-Wei Lee**[3] **Ee-Peng Lim**[1*]
[1]Singapore Management University
[2]Southwest Jiaotong University
[3]Singapore University of Technology and Design
[4]East China Normal University

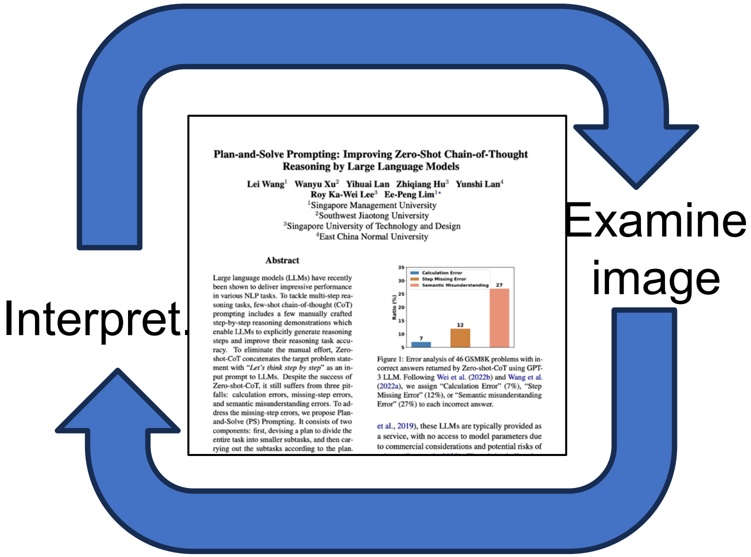
**Abstract**

Large language models (LLMs) have recently been shown to deliver impressive performance in various NLP tasks. To tackle multi-step reasoning tasks, few-shot chain-of-thought (CoT) prompting includes a few manually crafted step-by-step reasoning demonstrations which enable LLMs to explicitly generate reasoning steps and improve their reasoning task accuracy. To eliminate the manual effort, Zero-shot-CoT concatenates the target problem statement with "*Let's think step by step*" as an input prompt to LLMs. Despite the success of Zero-shot-CoT, it still suffers from three pitfalls: calculation errors, missing-step errors, and semantic misunderstanding errors. To address the missing-step errors, we propose Plan-and-Solve (PS) Prompting. It consists of two components: first, devising a plan to divide the entire task into smaller subtasks, and then carrying out the subtasks according to the plan.

Figure 1: Error analysis of 46 GSM8K problems with incorrect answers returned by Zero-shot-CoT using GPT-3 LLM. Following Wei et al. (2022b) and Wang et al. (2022a), we assign "Calculation Error" (7%), "Step Missing Error" (12%), or "Semantic misunderstanding Error" (27%) to each incorrect answer.

et al., 2019), these LLMs are typically provided as a service, with no access to model parameters due to commercial considerations and potential risks of

Input image



**Plan-and-Solve Prompting: Improving Zero-Shot Chain-of-Thought Reasoning by Large Language Models**

**Lei Wang**[1] **Wanyu Xu**[2] **Yihuai Lan** **Zhiqiang Hu**[3] **Yunshi Lan**[4]
**Roy Ka-Wei Lee**[3] **Ee-Peng Lim**[1*]
[1]Singapore Management University
[2]Southwest Jiaotong University
[3]Singapore University of Technology and Design
[4]East China Normal University

**Abstract**

Large language models (LLMs) have recently been shown to deliver impressive performance in various NLP tasks. To tackle multi-step reasoning tasks, few-shot chain-of-thought (CoT) prompting includes a few manually crafted step-by-step reasoning demonstrations which enable LLMs to explicitly generate reasoning steps and improve their reasoning task accuracy. To eliminate the manual effort, Zero-shot-CoT concatenates the target problem statement with "*Let's think step by step*" as an input prompt to LLMs. Despite the success of Zero-shot-CoT, it still suffers from three pitfalls: calculation errors, missing-step errors, and semantic misunderstanding errors. To address the missing-step errors, we propose Plan-and-Solve (PS) Prompting. It consists of two components: first, devising a plan to divide the entire task into smaller subtasks, and then carrying out the subtasks according to the plan.

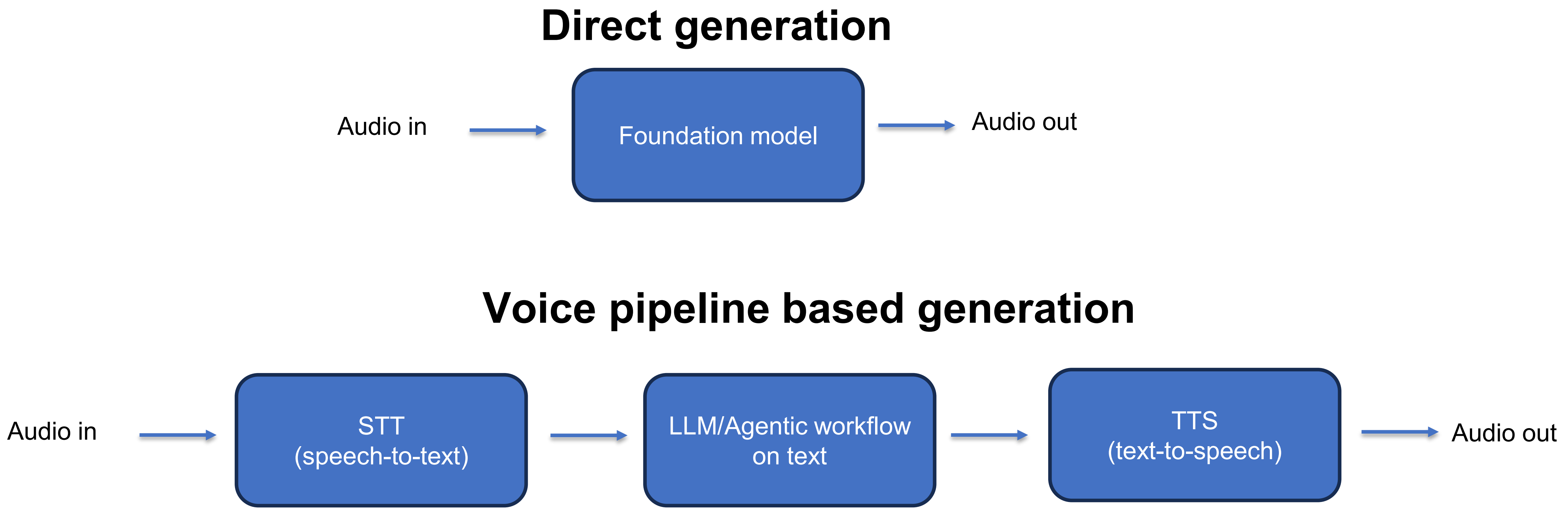
Figure 1: Error analysis of 46 GSM8K problems with incorrect answers returned by Zero-shot-CoT using GPT-3 LLM. Following Wei et al. (2022b) and Wang et al. (2022a), we assign "Calculation Error" (7%), "Step Missing Error" (12%), or "Semantic misunderstanding Error" (27%) to each incorrect answer.

et al., 2019), these LLMs are typically provided as a service, with no access to model parameters due to commercial considerations and potential risks of

Andrew Ng

# 4. Voice stack

- Building voice applications is much easier than a year ago.
- Voice stack options:



Andrew Ng

# 5. Data engineering

- Data engineering's importance is rising, particularly on management of text, images, video, audio (also called unstructured data)
- Getting the data to AI for processing has high value

Data gravity is decreasing.

- Data Gravity is the idea that data tends to attract other data and compute. You would not transmit 1TB of data across clouds for processing.
- But for GenAI workloads, transmission costs are dwarfed by processing costs:
    - 1GB of data
    - Cost of transmission: ~$0.10
    - Cost of processing: ~$30-40 (gpt-4o-mini prices)
- This is leading to highly distributed software architectures, where we bring together many best-of-breed AI services.

# Talent gap

- Shortage of skilled AI engineers continues
- Easily 10x difference in productivity between skilled and less skilled AI engineers.

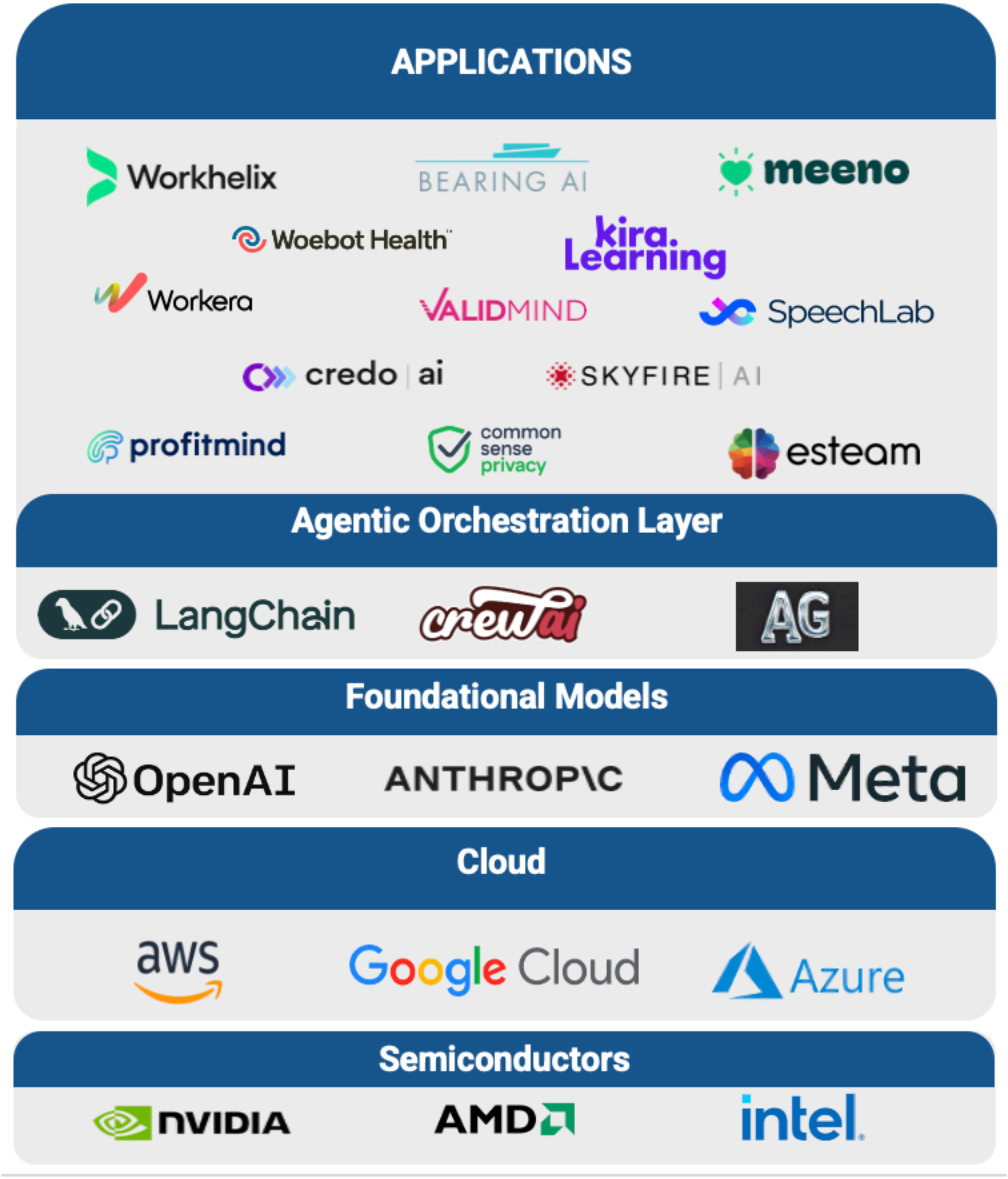
Kirsty Tan
AI Aspire

# Five important technical AI trends

Technology trends are making it easier to identify and build AI applications.

1. **AI coding assistance.** This is software development much more pervasive and efficient .
2. **Fast prototyping.** Generative AI is making it possible to build AI prototypes very efficiently. This is changing how we innovate and invent new things.
3. **Visual AI**. The text processing revolution has arrived. The image processing revolution is coming, and will enable many new visual applications in manufacturing, self-driving, etc.
4. **Voice stack**. There will be many compelling voice-based applications.
5. **Data engineering**'s importance is rising, particularly on management of unstructured data (text, images, video, audio).

# Conclusion



APPLICATIONS

Agentic Orchestration Layer

Foundational Models

Cloud

Semiconductors

This is a wonderful time to build!

# END