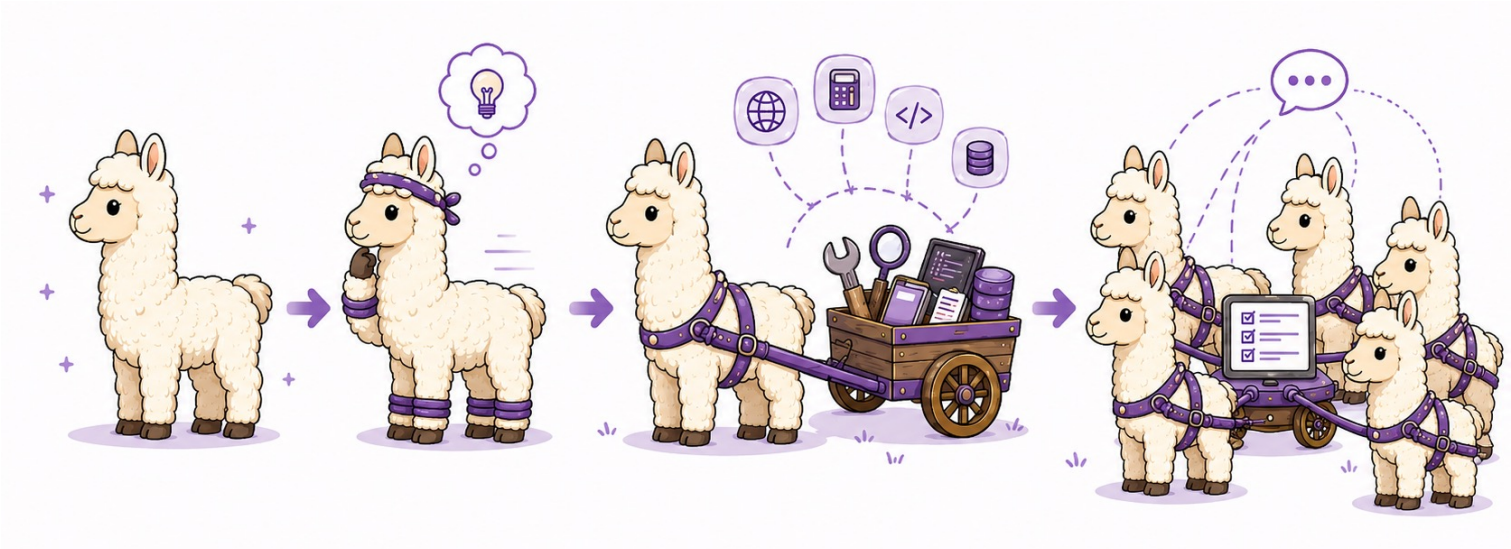


From Tool-Using LLMs to Orchestrated Agent Systems

AutoML Summer School · Elisabeth Kirsten · 2026

About me

Elisabeth Kirsten

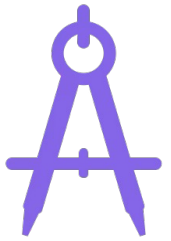
Doctoral Researcher

- Ruhr University Bochum · RC Trust · AI & Society Lab
- **Research:**
  - LLM agents
  - Responsible & Human-centric AI
  - AI evaluations
- **Today:** Intro to Agents

LLMs Can Perform Incredible Feats

“Traditional” NLP Tasks
(classification, summarization, etc.)

Poetry



Maths

Can you write a proof that there are infinitely many primes, with every line that rhymes?

GPT-4:
Yes, I think I can,
Though it might take a clever plan.
I’ll start by noting **Euclid’s proof**,
Which shows that primes aren’t just aloof.
Assume we have a finite list,
Of primes, and that none have been missed.
…
So we’ve found a **contradiction**,
…

[Bubeck et al.]

What Can LLMs Still Not Do?

Find three recent papers on LLM Agents and tell me which one is most relevant for our course.

Model may need to…

- ❑ Search the web
- ❑ Open several sources
- ❑ Decide which sources are relevant
- ❑ Compare different sources
- ❑ Maybe search again with better keywords

What Can LLMs Still Not Do?

Find a train from Bochum to Berlin tomorrow morning, under 80EUR, arriving before noon.

Model may need to…

- ❑ Query a live source
- ❑ Filter options
- ❑ Compare constraints
- ❑ Revise the search
- ❑ Ask before booking

What Can LLMs Still Not Do?

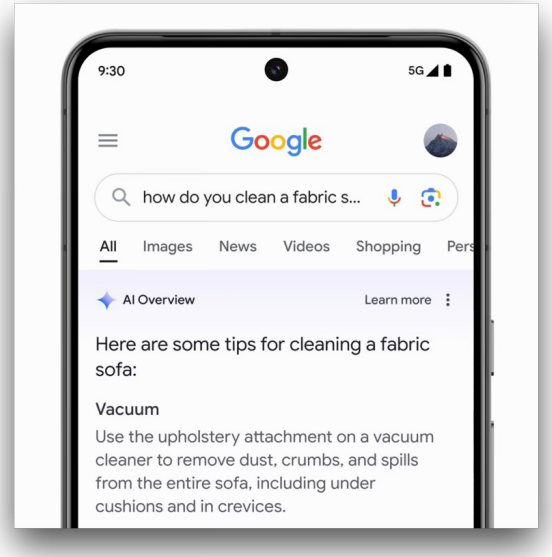
Fix this failing test.

Model may need to…

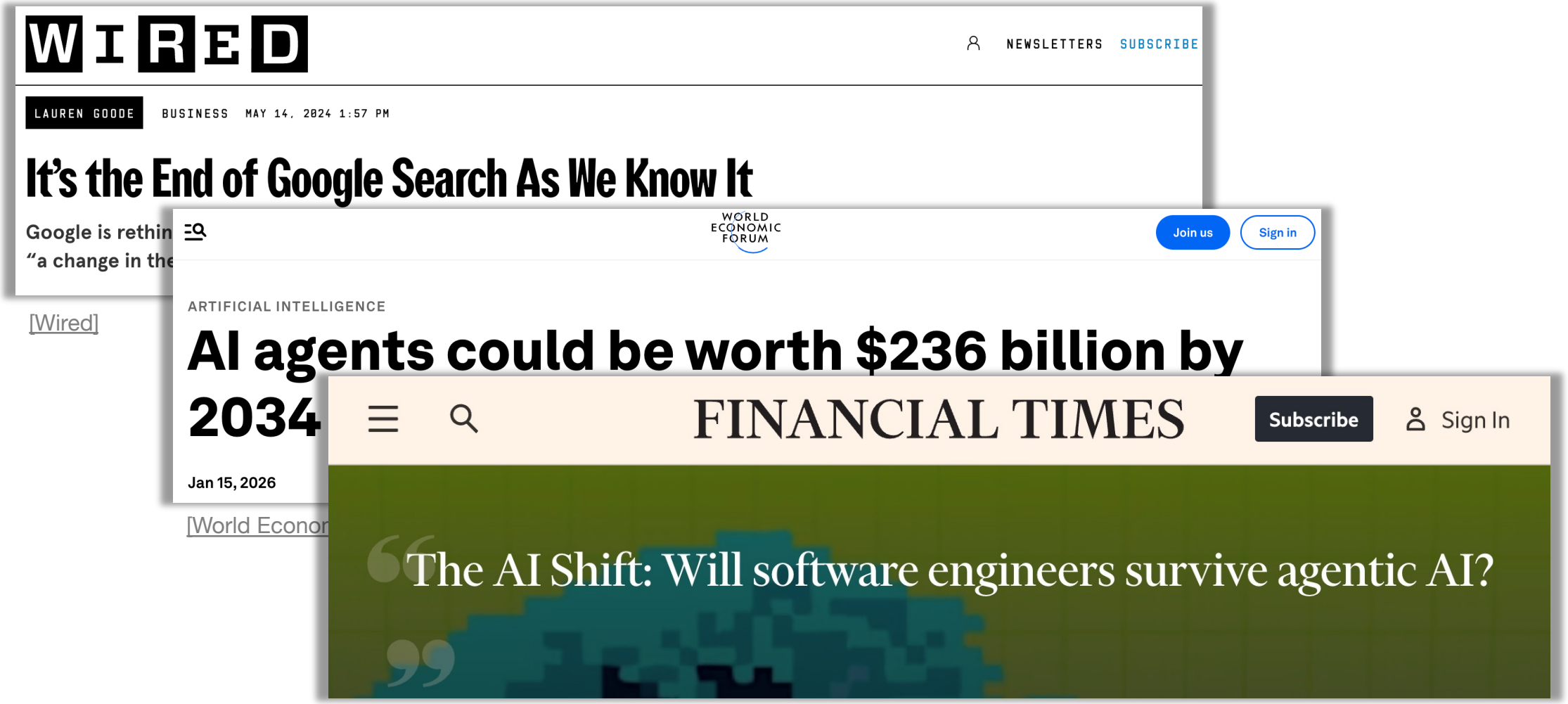
- ❑ Inspect files
- ❑ Run tests
- ❑ Observe errors
- ❑ Edit code
- ❑ Run tests again
- ❑ Stop when tests pass

Answer Generation ≠ Task Completion

Agents Everywhere, All At Once



Immense Real-World Impact



[Wired]

[World Econor

[Financial Times]

In This Tutorial

- **Part 1: Introduction to LLM Agents**
- **Part 2: Orchestration & Multi-agent Systems**

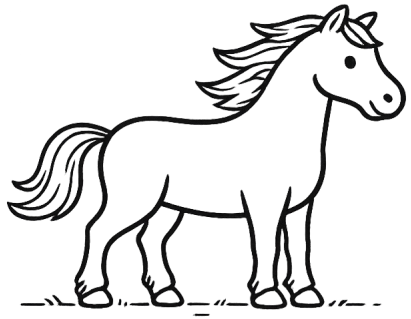
+ Exercises

Key Learnings:

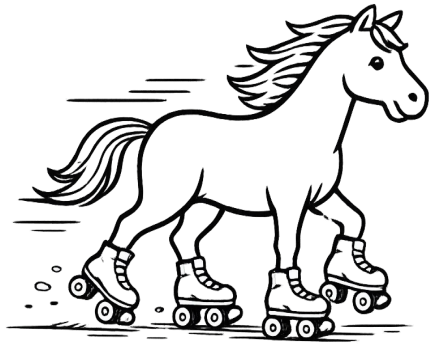
1. **Define & build** agents
2. **Analyze** multi-agent systems
3. **Think critically** about the agentic AI economy

Part 1: Intro to Agents

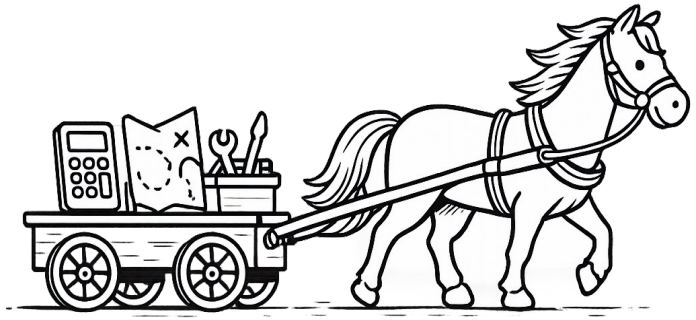
A Brief History of Agents



Conventional LLM



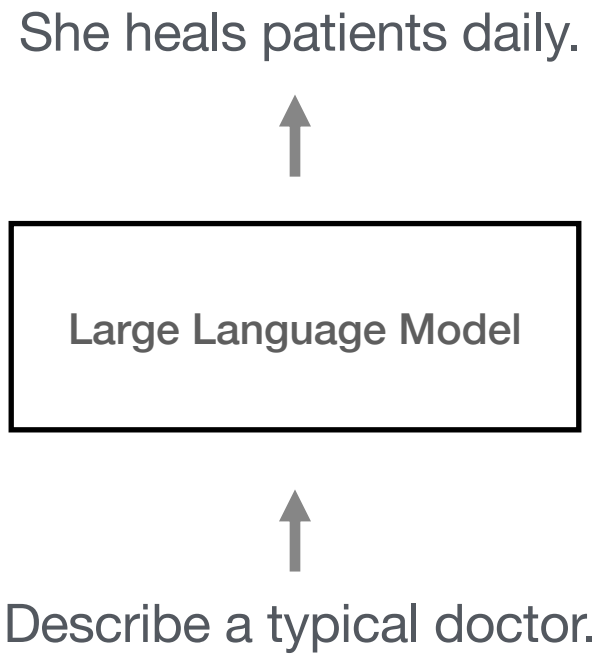
Reasoning LLM



LLM in agent harness

- **Conventional LLM:** Prompt → Answer
- **Reasoning LLM:** Query → Think → Answer
- **Agentic System:** ???

(Rough) Anatomy of a generation

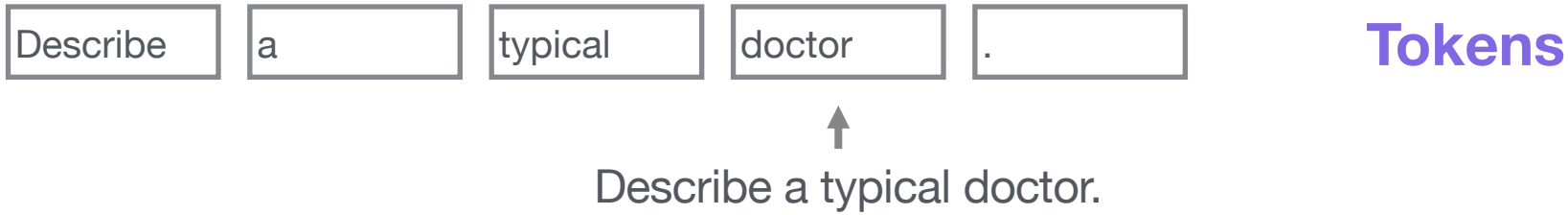


*Fictitious Example

(Rough) Anatomy of a generation

Describe a typical doctor.

Step 1: Tokenization



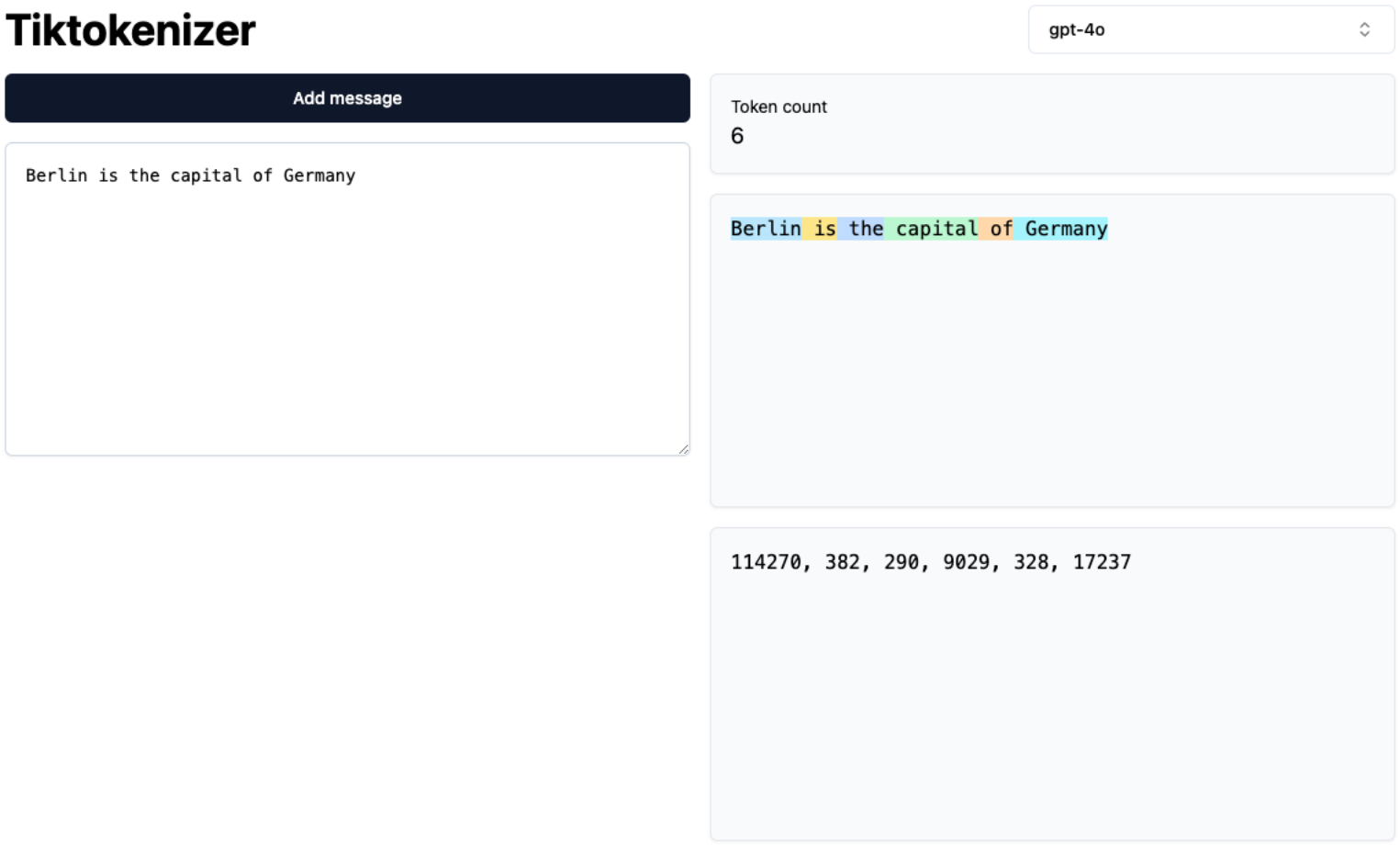
Tokenization

- Split the input text into individual tokens (the *“atoms”* of LLMs)
- A token is usually smaller than a word, e.g., hopeful → hope + ful
- Helps models handle complex languages and larger vocabularies more efficiently

Tokenization

- Words share subparts, e.g., consider the 7 words with 2 variations (27 words in total)
  - color, hope, help, harm, lust, mean, power
  - colorful, hopeful, helpful, harmful, lustful, meaningful, powerful
  - coloring, hoping, helping, harming, lusting, meaning, powering
- With ful and ing as subwords, we can represent all words as 7+2 = 9 tokens instead of 27 words

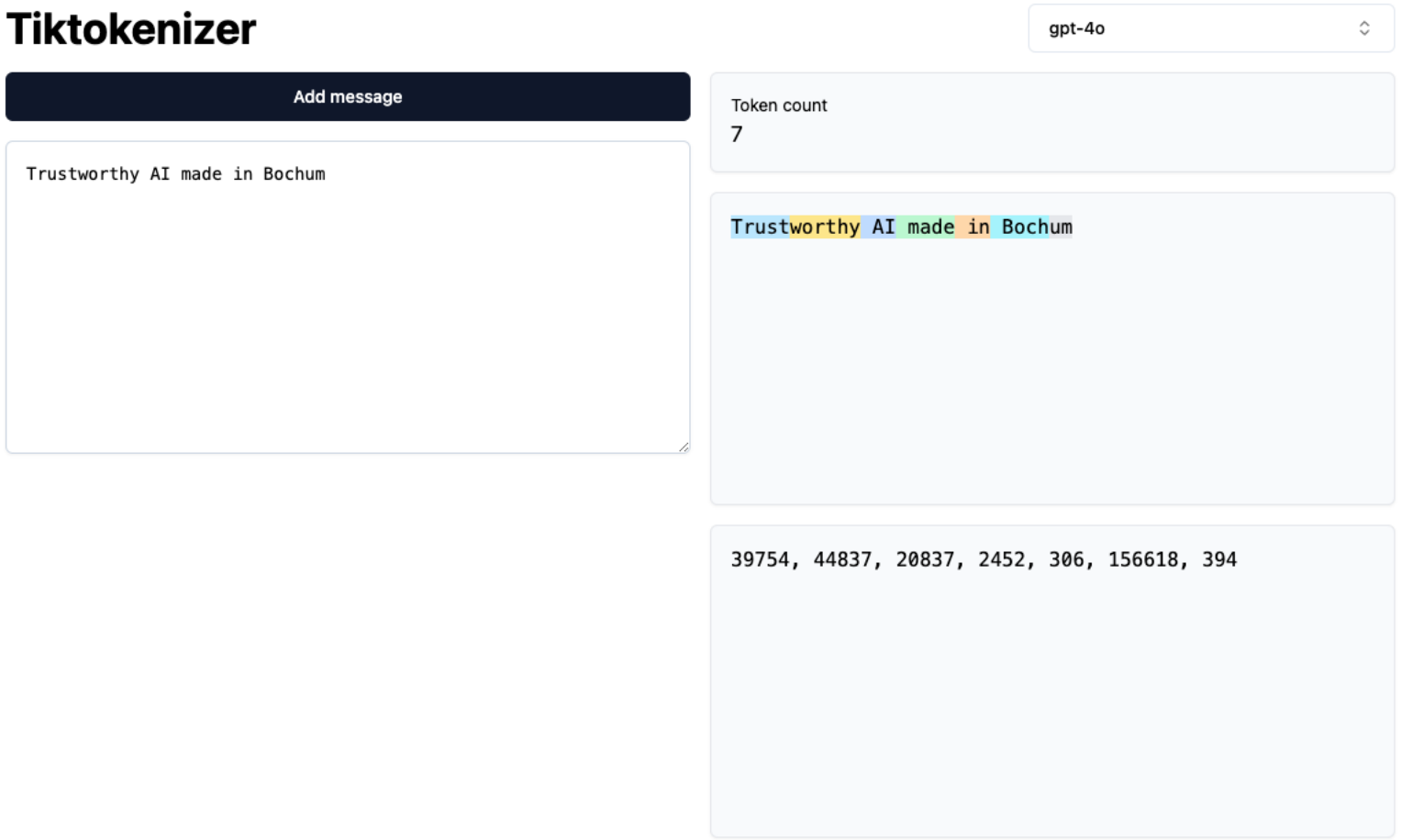
Tokenization



[Tiktokenizer]

Each word corresponds to a token

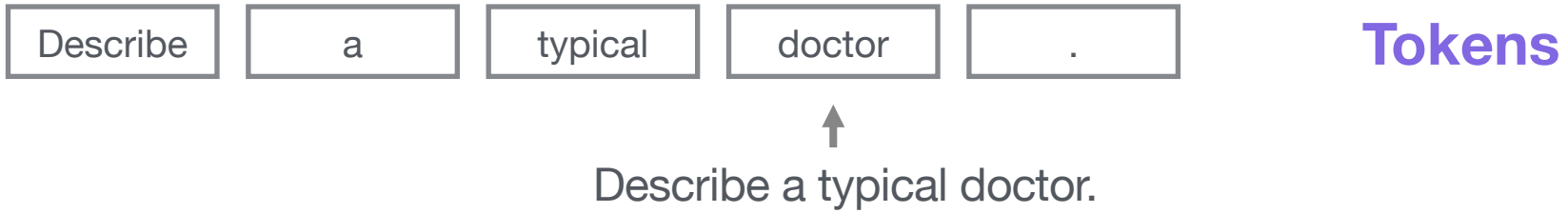
Tokenization



[Tiktokenizer]

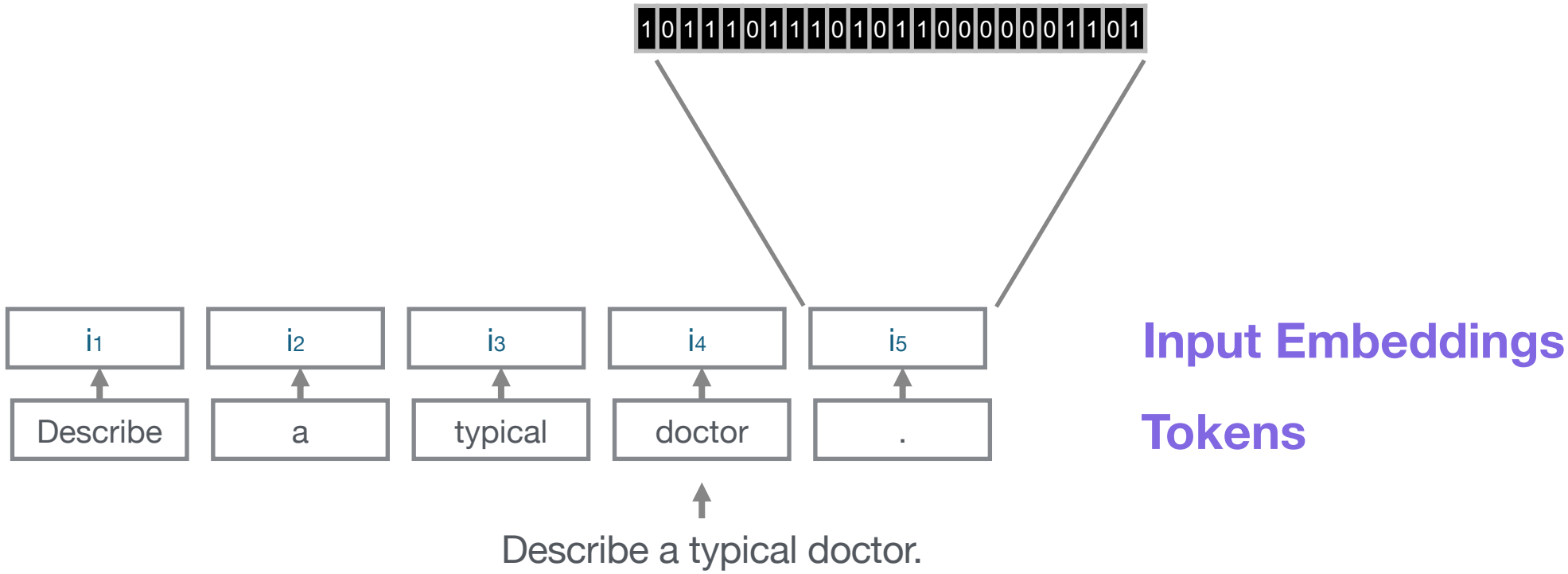
Some words are split into multiple tokens

Step 1: Tokenization

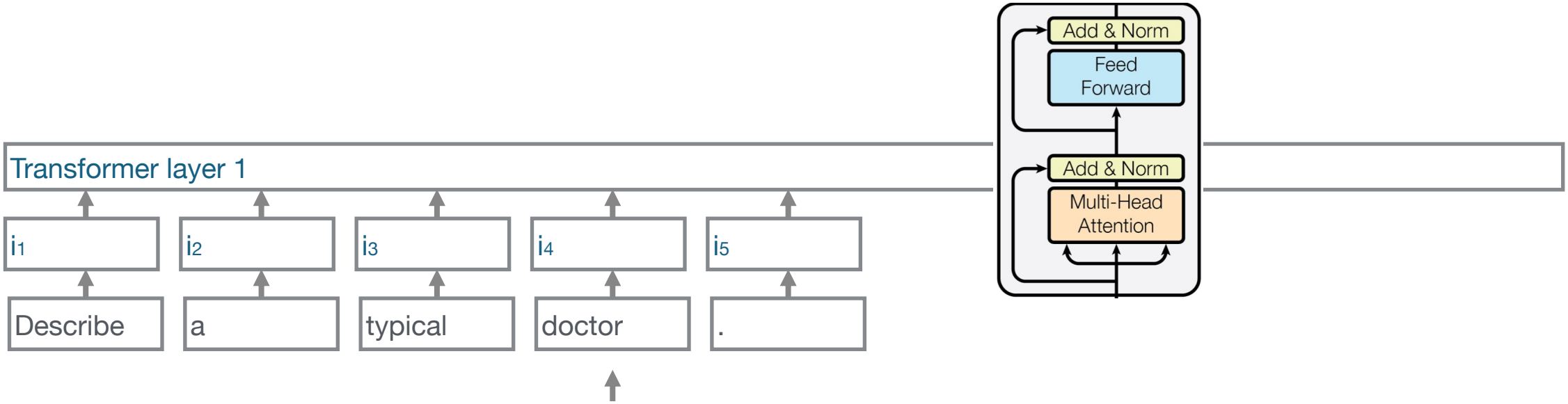


Step 2: Conversion to input embeddings

- Models work with numbers instead of text
- Map each token to a unique vector (Embedding Lookup)
- Capture semantic meaning & contextual understanding

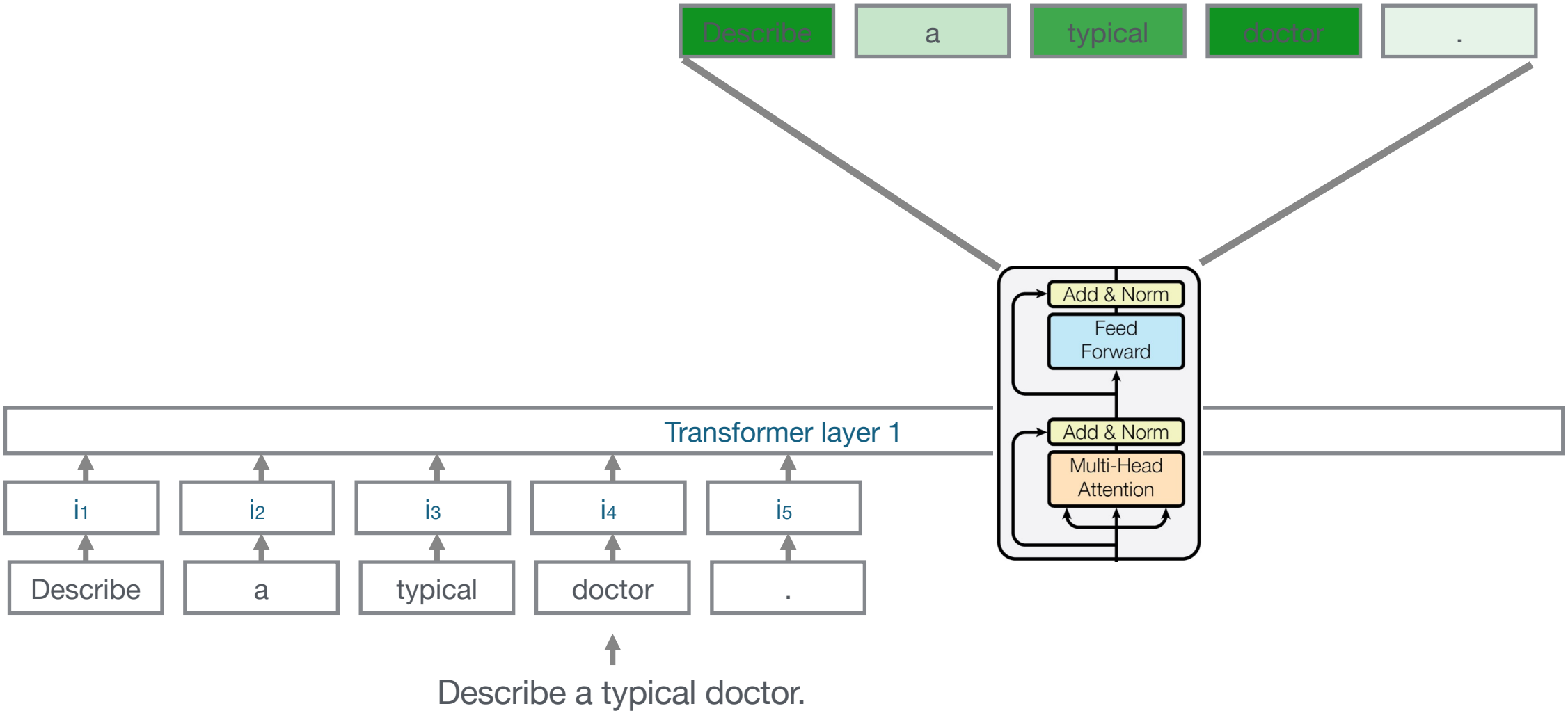


Step 3: Self-attention

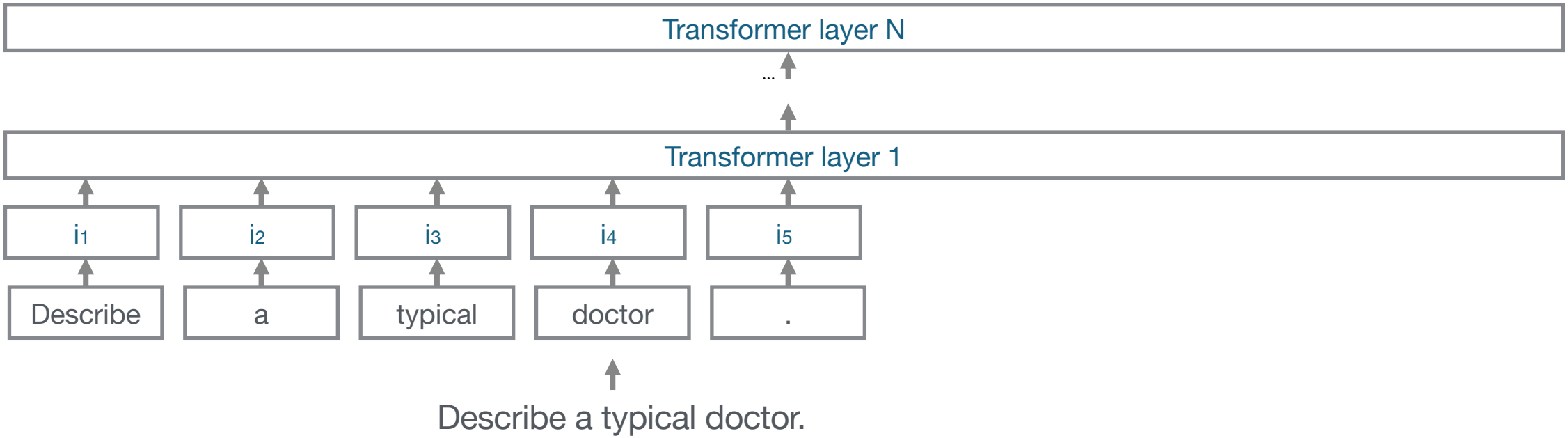


[Figure: Vaswani et al]

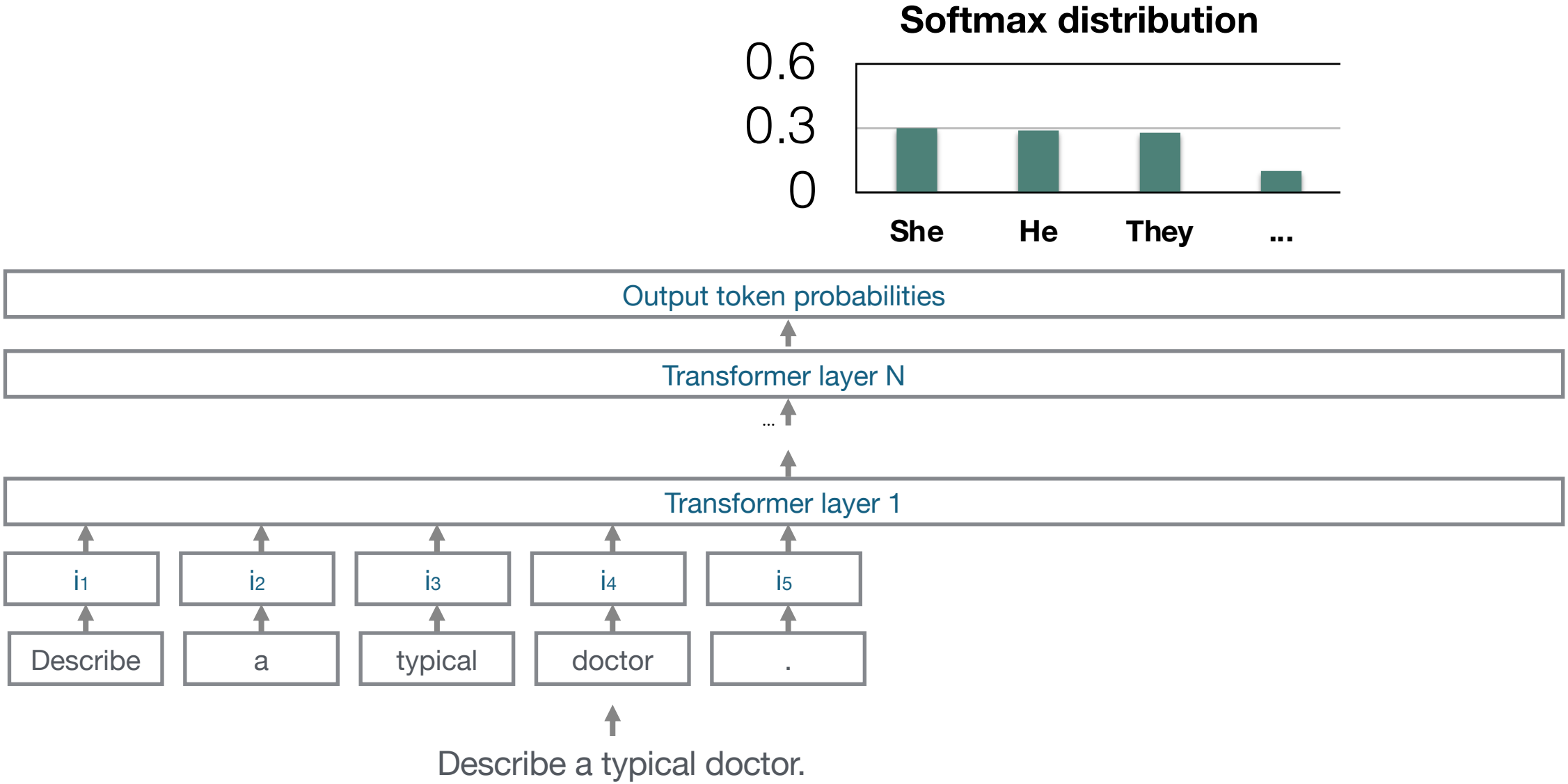
Step 3: Self-attention



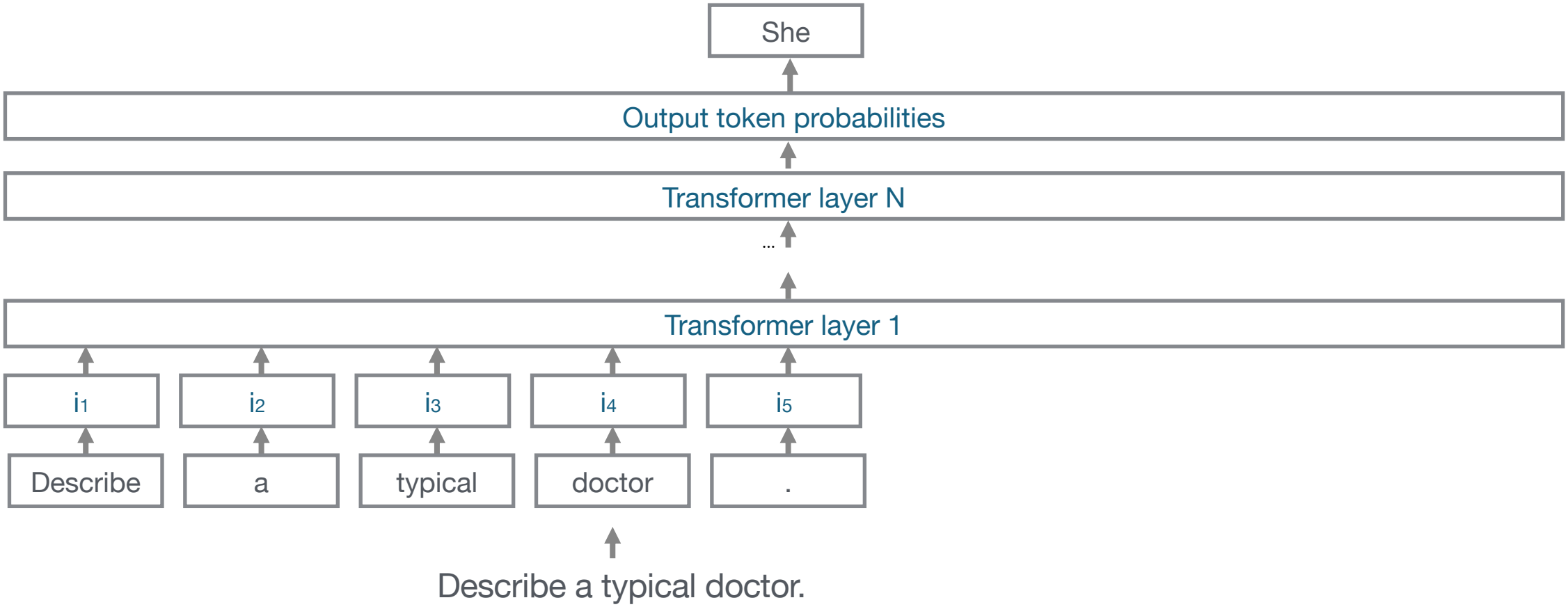
Step 3: Self-attention



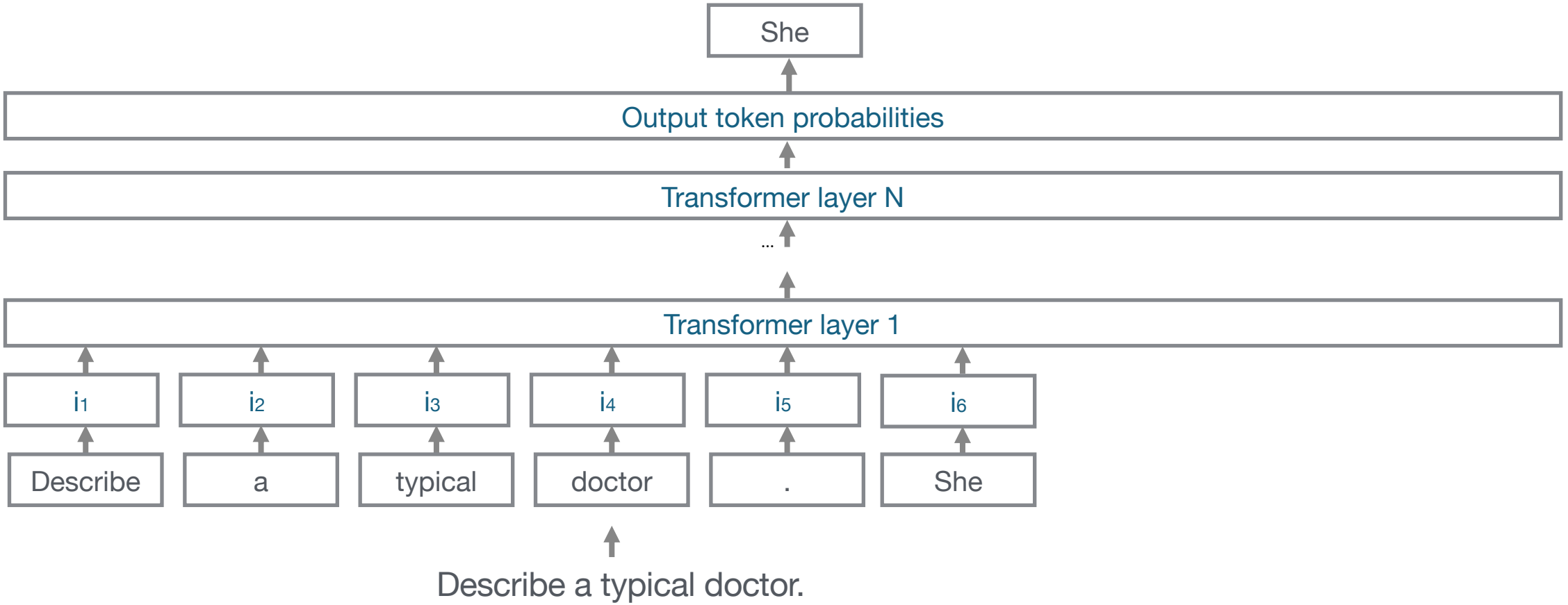
Step 4: Probability of the next token



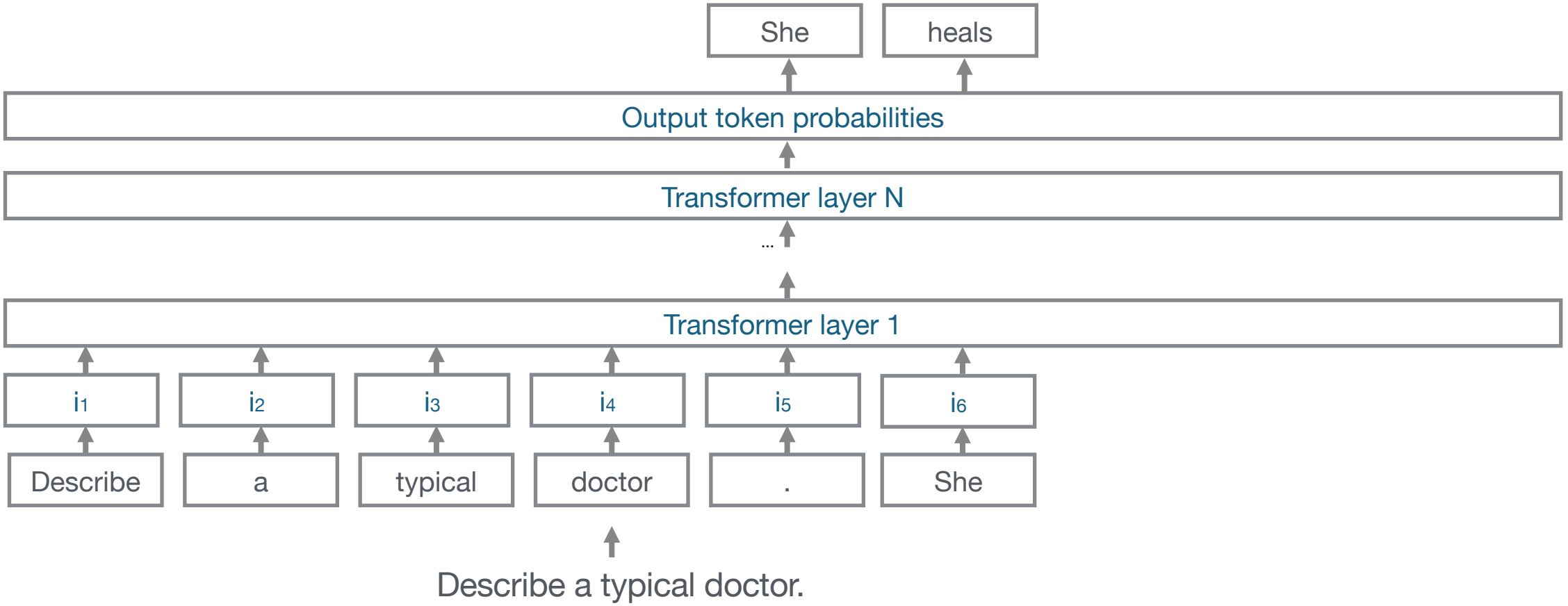
Step 5: Generate the token



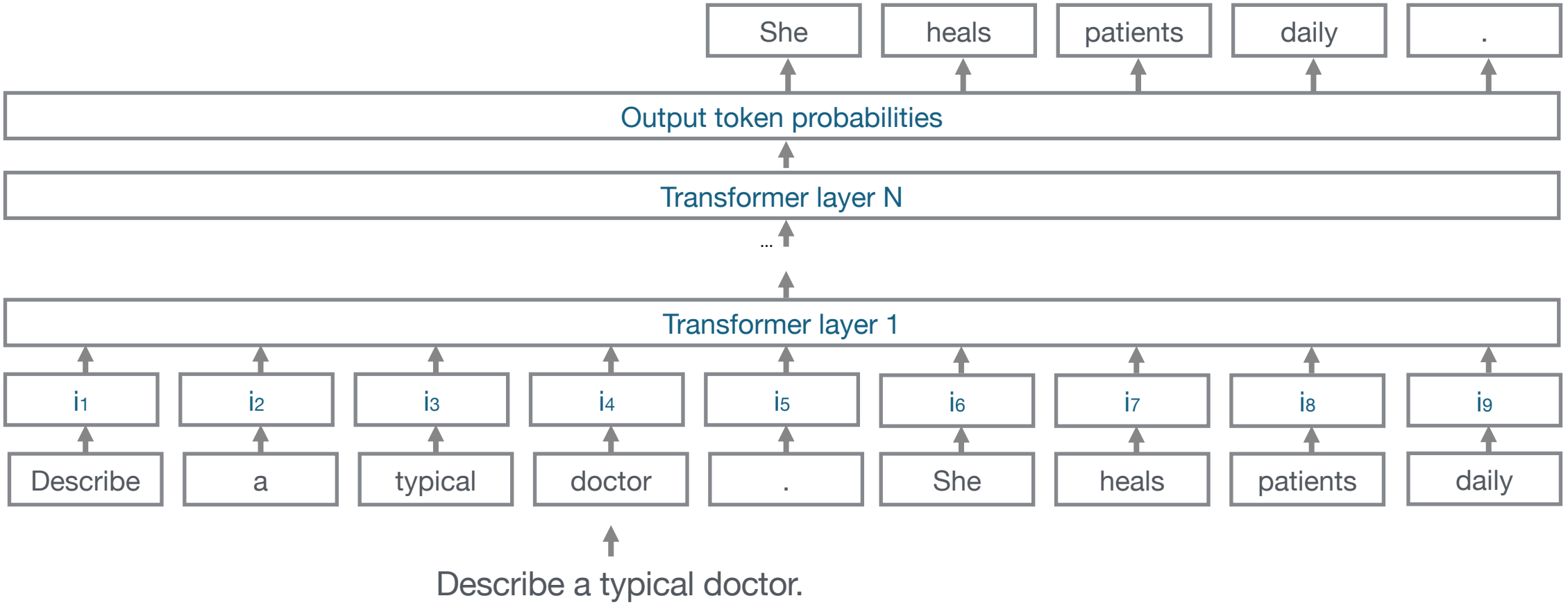
Step 6: Add the token to the input



Continue …


She
heals
Output token probabilities
Transformer layer N
...
Transformer layer 1
i1
i2
i3
i4
i5
i6
Describe
a
typical
doctor
.
She
Describe a typical doctor.

Until some stopping condition is met

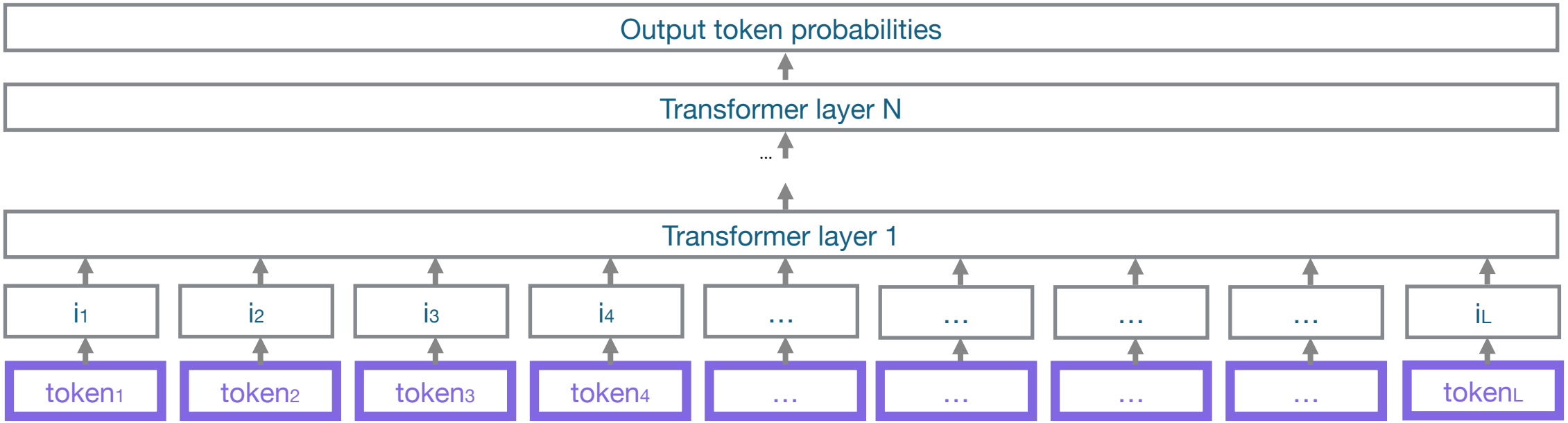


Stopping conditions

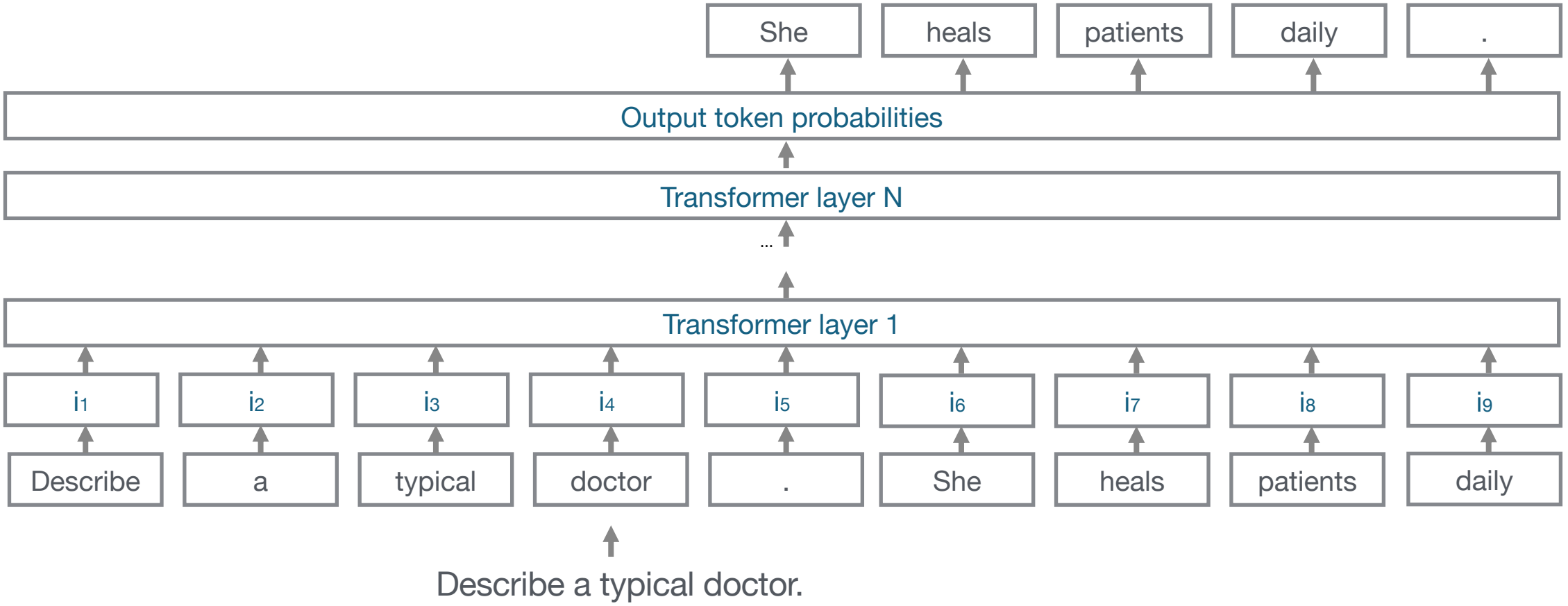
- Generate only 10 new tokens
- Stop when the model generates a specific token, e.g., fullstop “.”

Transformers have a maximum sequence length

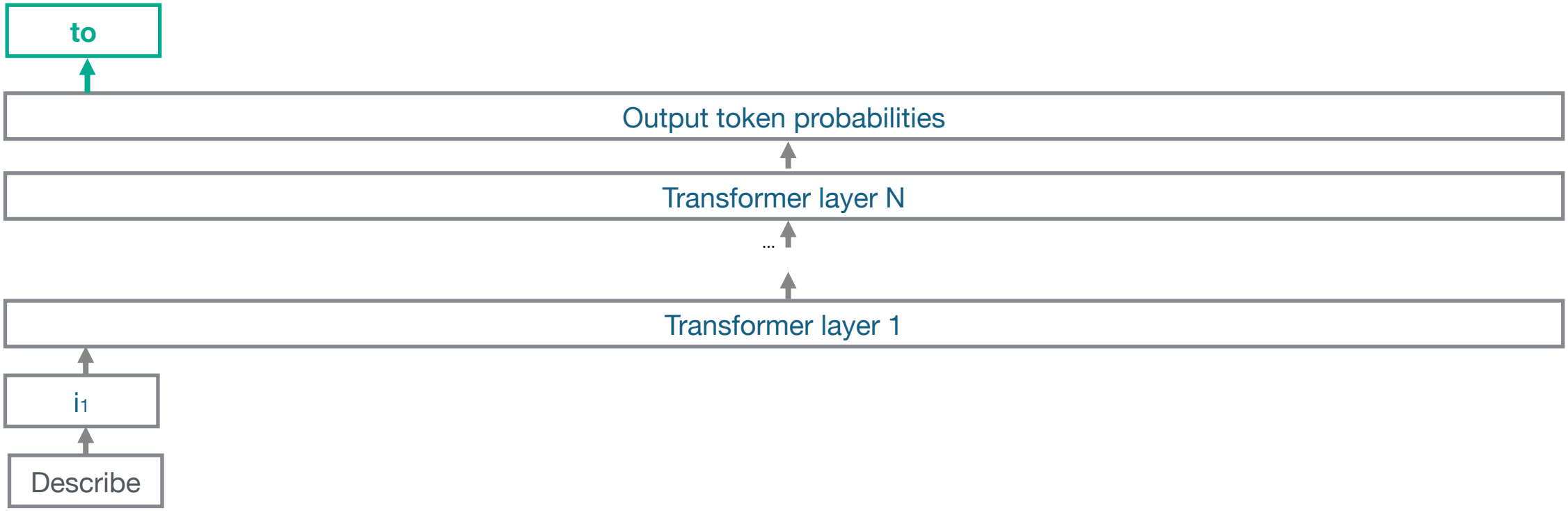
Sequence length = L



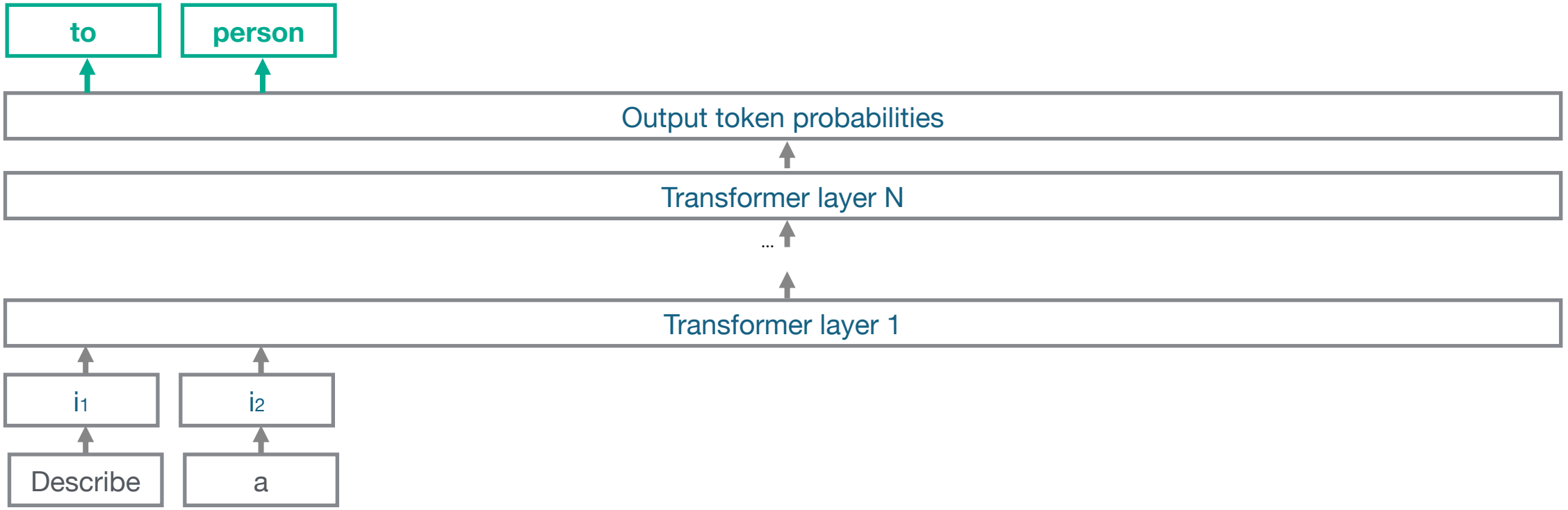
Modern LLMs: A prediction following every input location



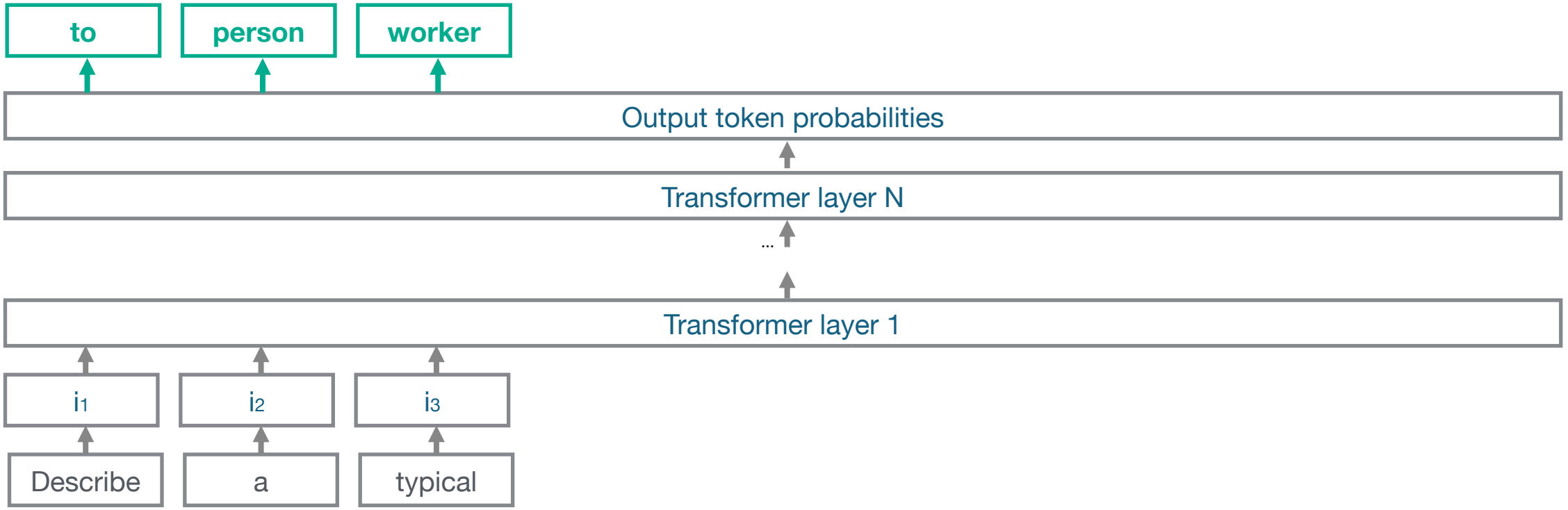
Modern LLMs: A prediction following every input location



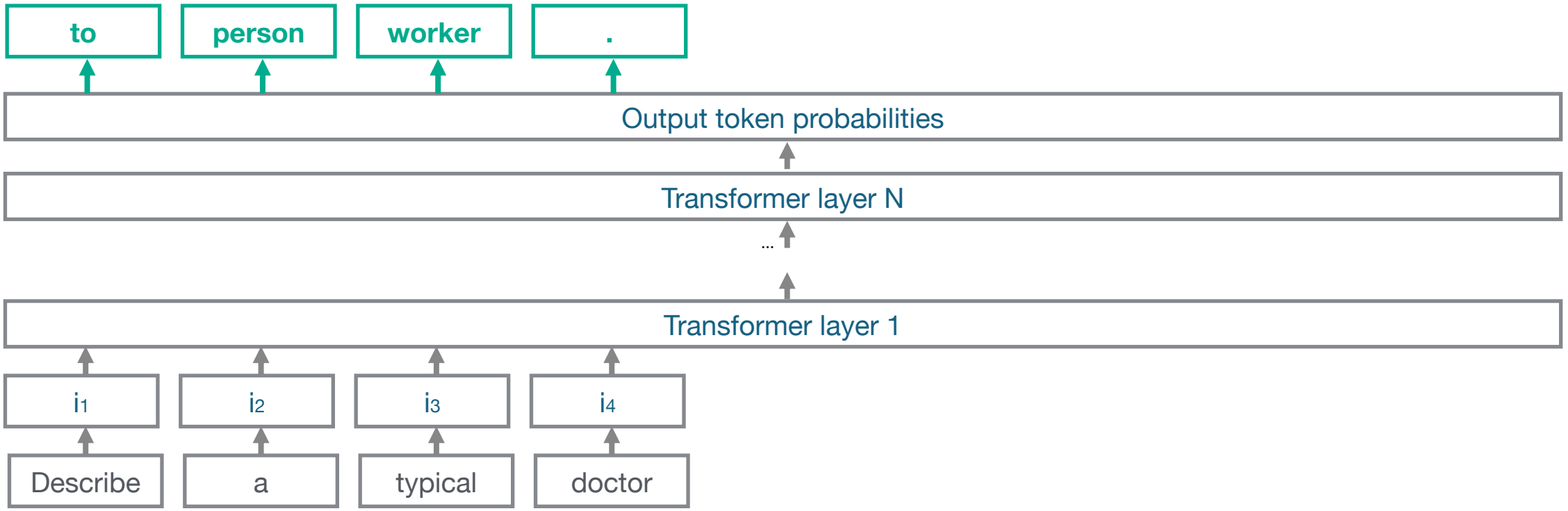
Modern LLMs: A prediction following every input location



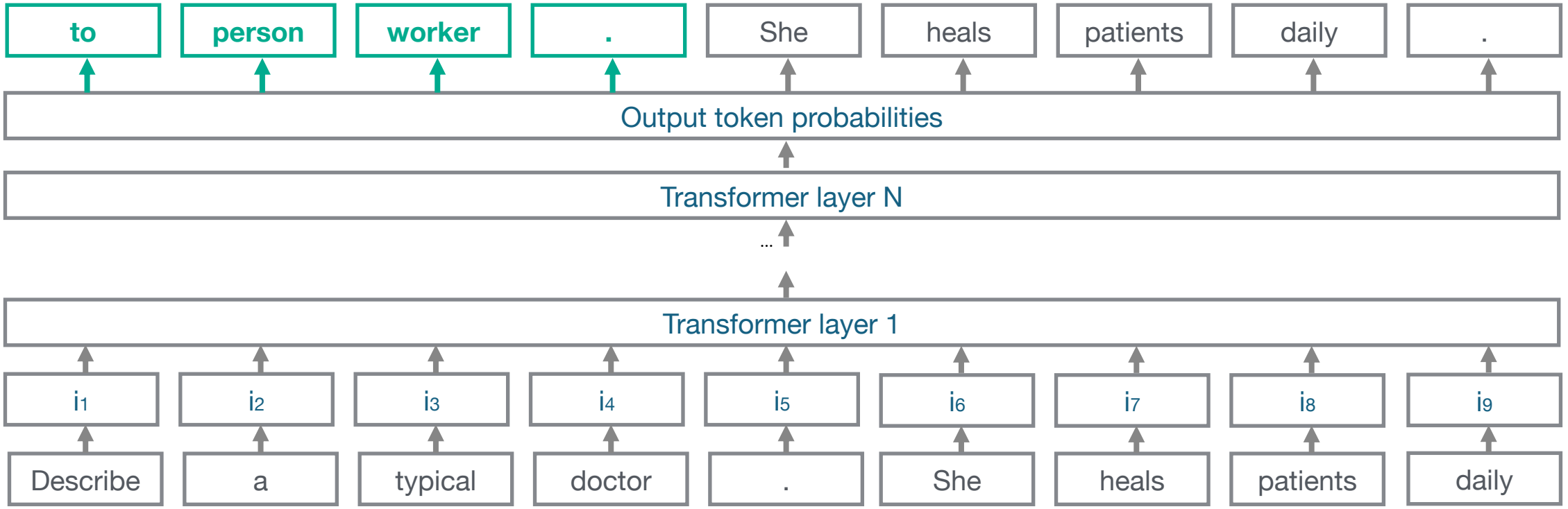
Modern LLMs: A prediction following every input location



Modern LLMs: A prediction following every input location

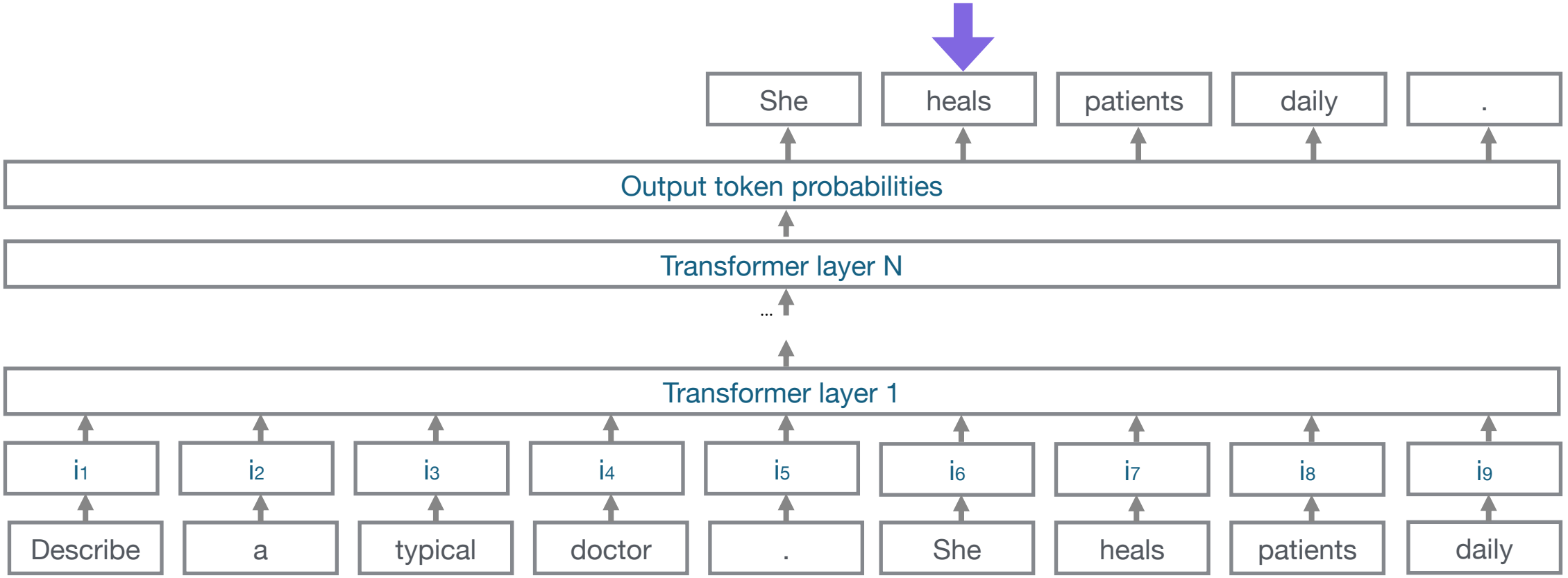


Modern LLMs: A prediction following every input location



Most modern LLMs are causal

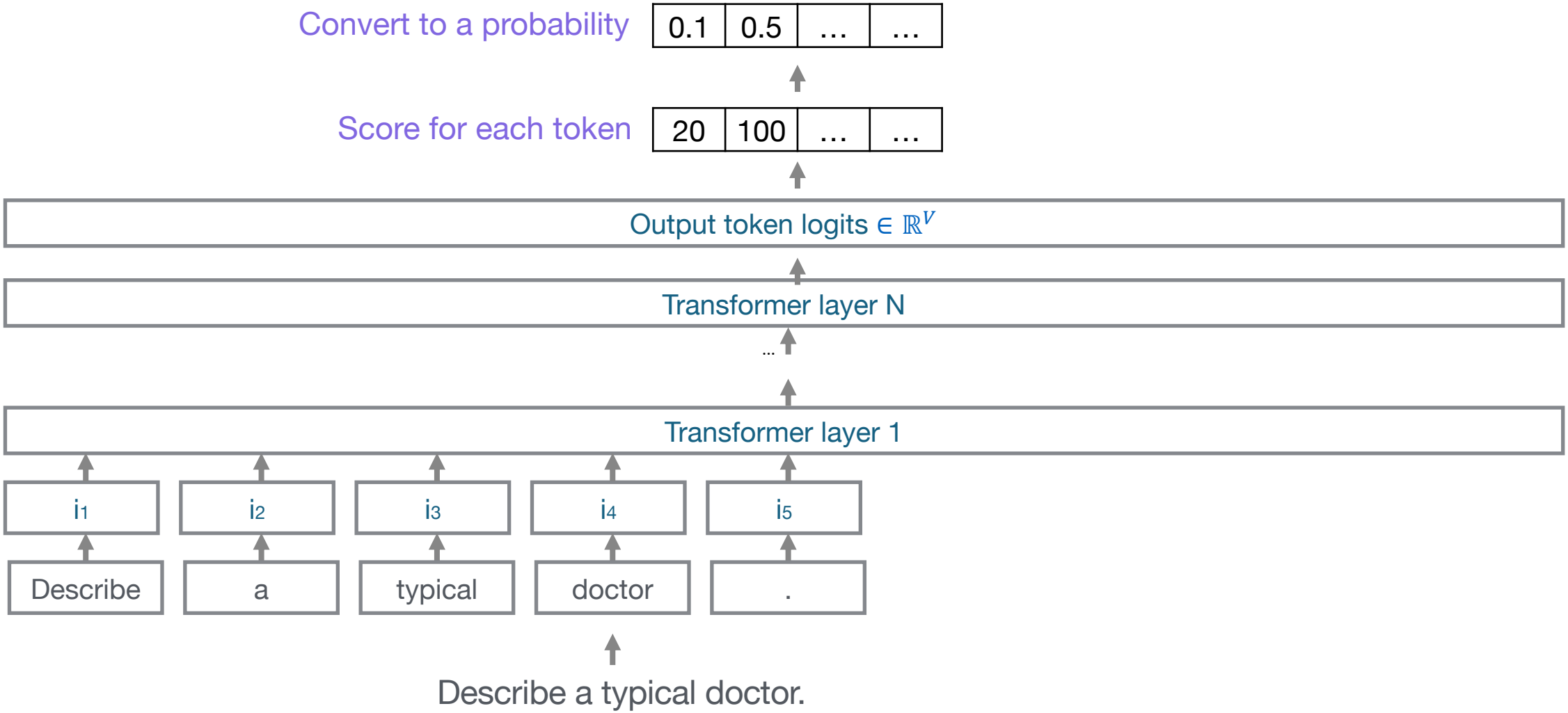
- Model can only look left (in the past)
- Cannot look right (in the future)



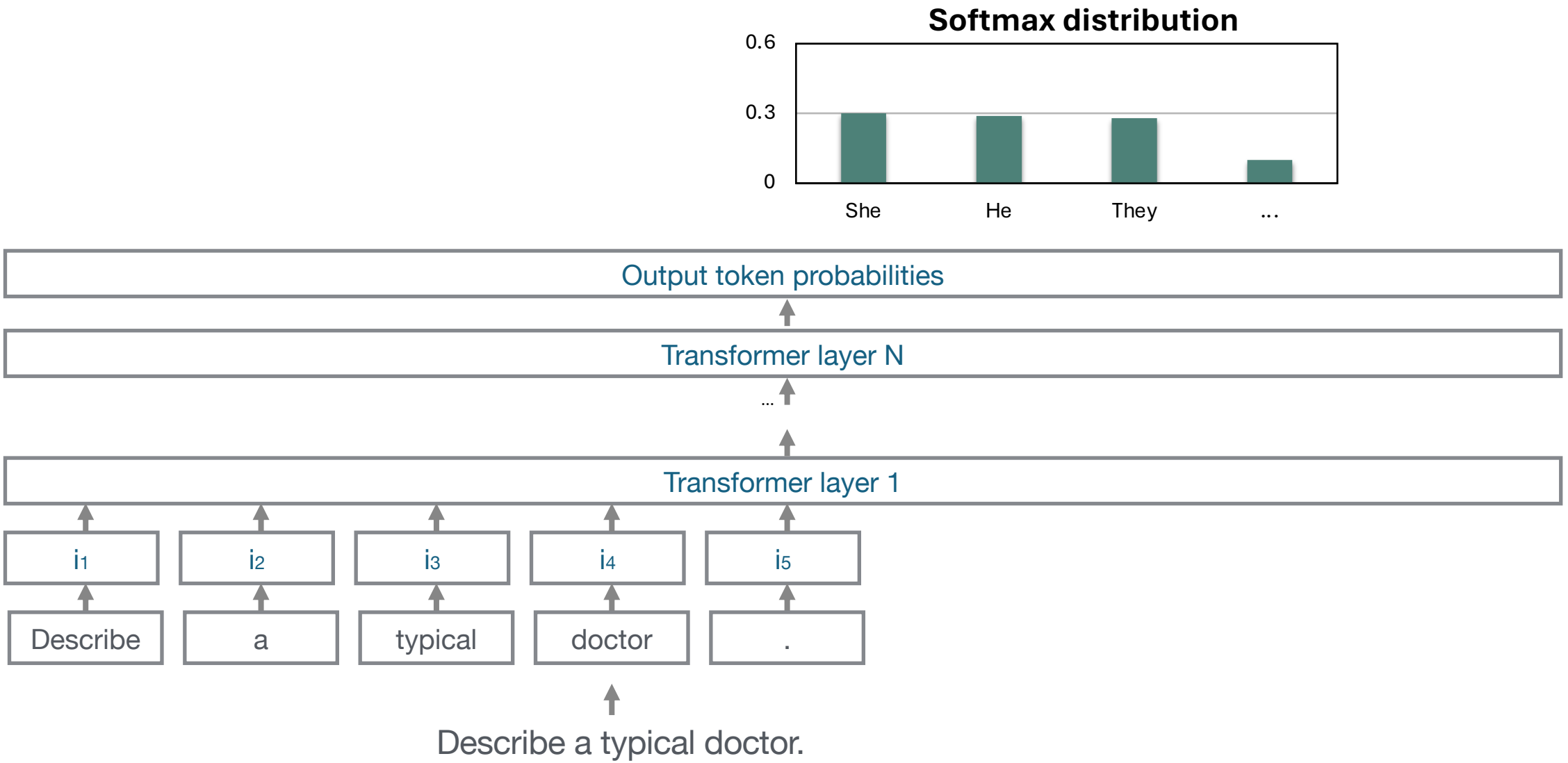
Most modern LLMs are causal

- Model can only look left (in the past)
- Cannot look right (in the future)

Selecting the token to generate



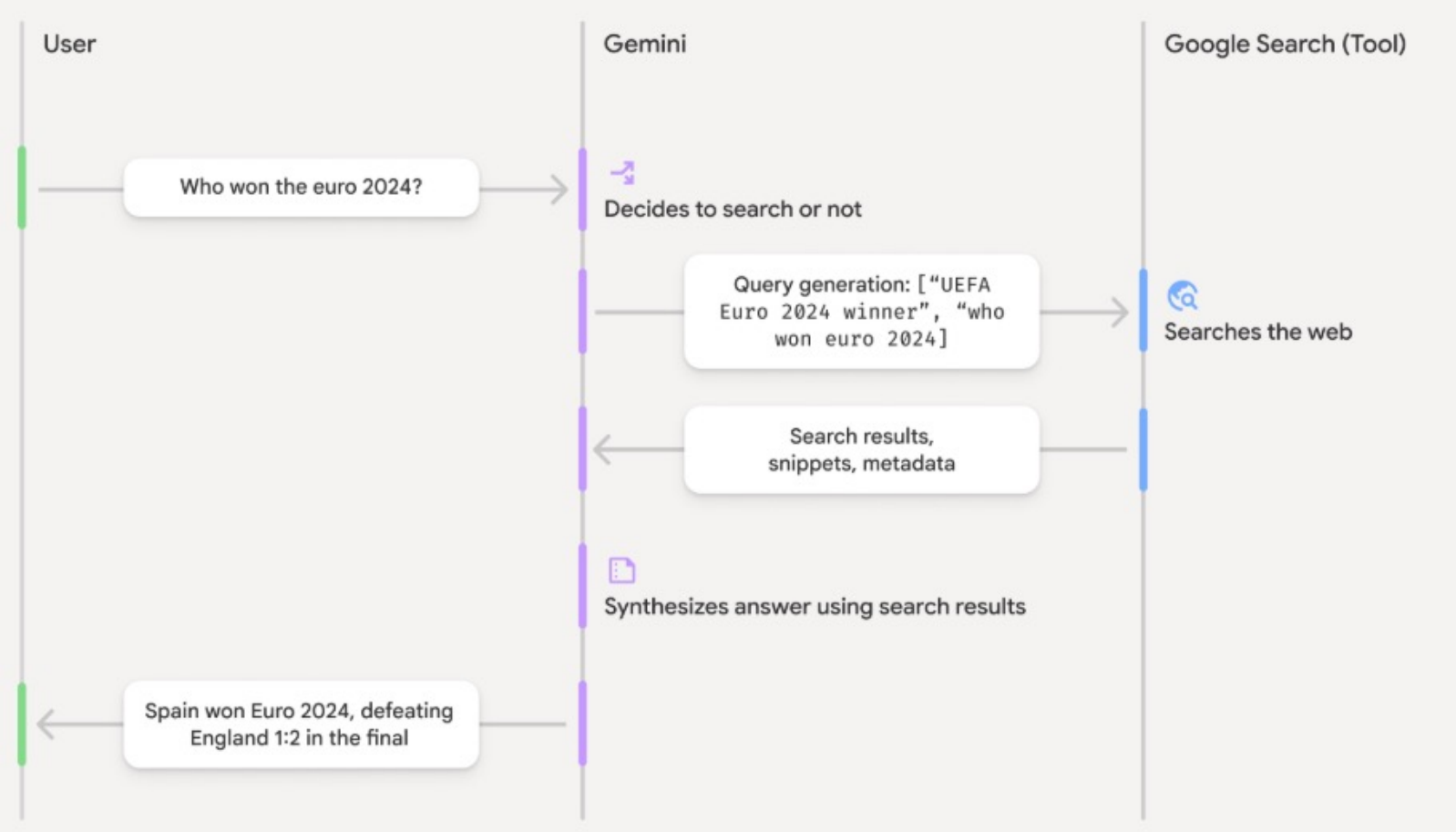
Selecting the token to generate



What Are Agents?

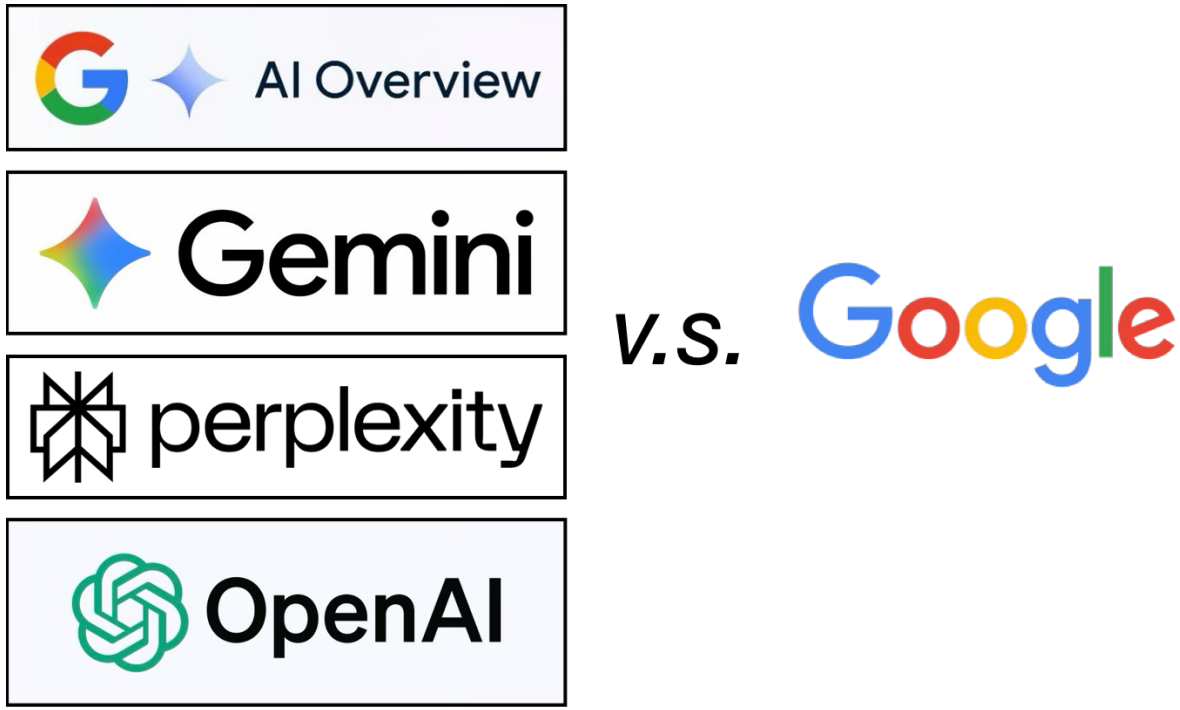
Agents:
LLMs with access to tools and the freedom to use them to get things done.

Web Search Agents



Grounding with Google Search

Many LLMs now support Web Search



"Characterizing Web Search in The Age of Generative AI"

Where Does Knowledge Come From?

Internal Knowledge

Learned during training

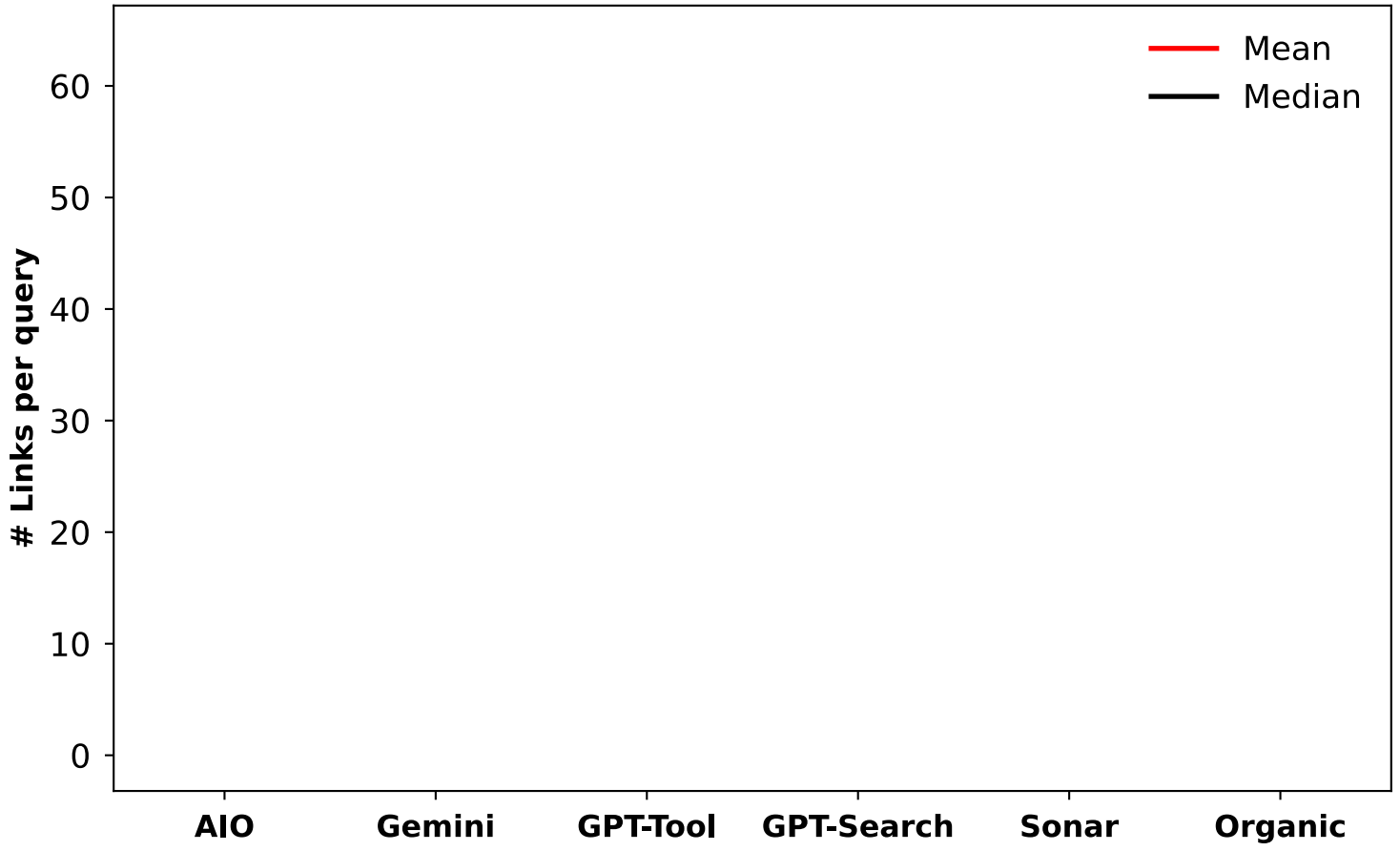
- Paris is in France
- Python syntax
- General concepts

External Knowledge

Retrieved / produced at inference

- Today's weather
- FIFA world cup winner
- Uploaded documents

Internal vs. External Knowledge



Models rely on external information to varying degrees

Internal vs. External Knowledge

Query: “Ricky Hatton Cause of Death”

Ricky Hatton, the former professional boxer, **is alive.** There might be some confusion or misinformation, as of the latest information, he has not passed away.

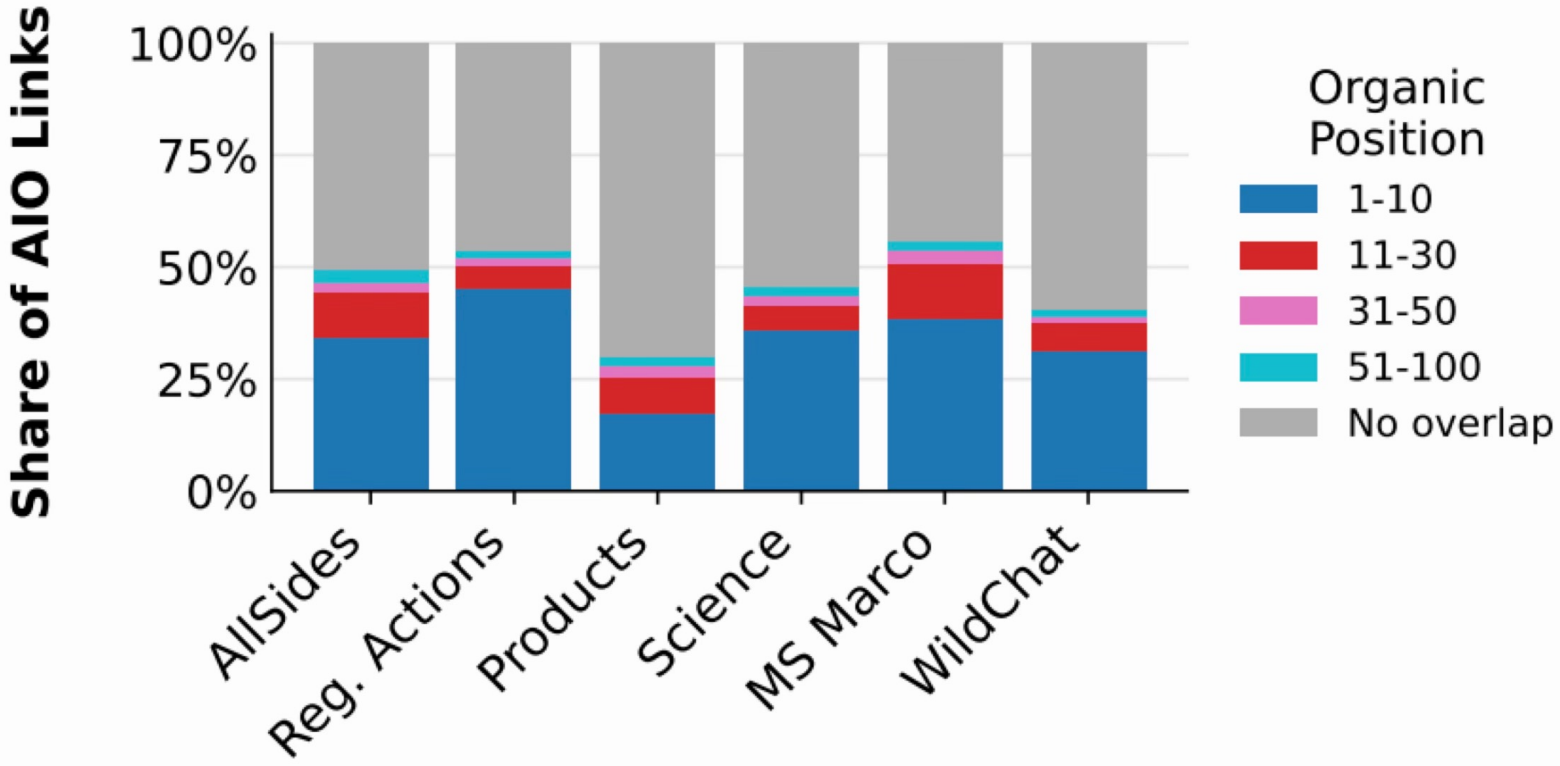
GPT-Tool

Former world boxing champion Ricky Hatton **has died at the age of 46.** His body was found by Manchester Police[…].

Gemini

Relying on internal knowledge only can affect the accuracy of results

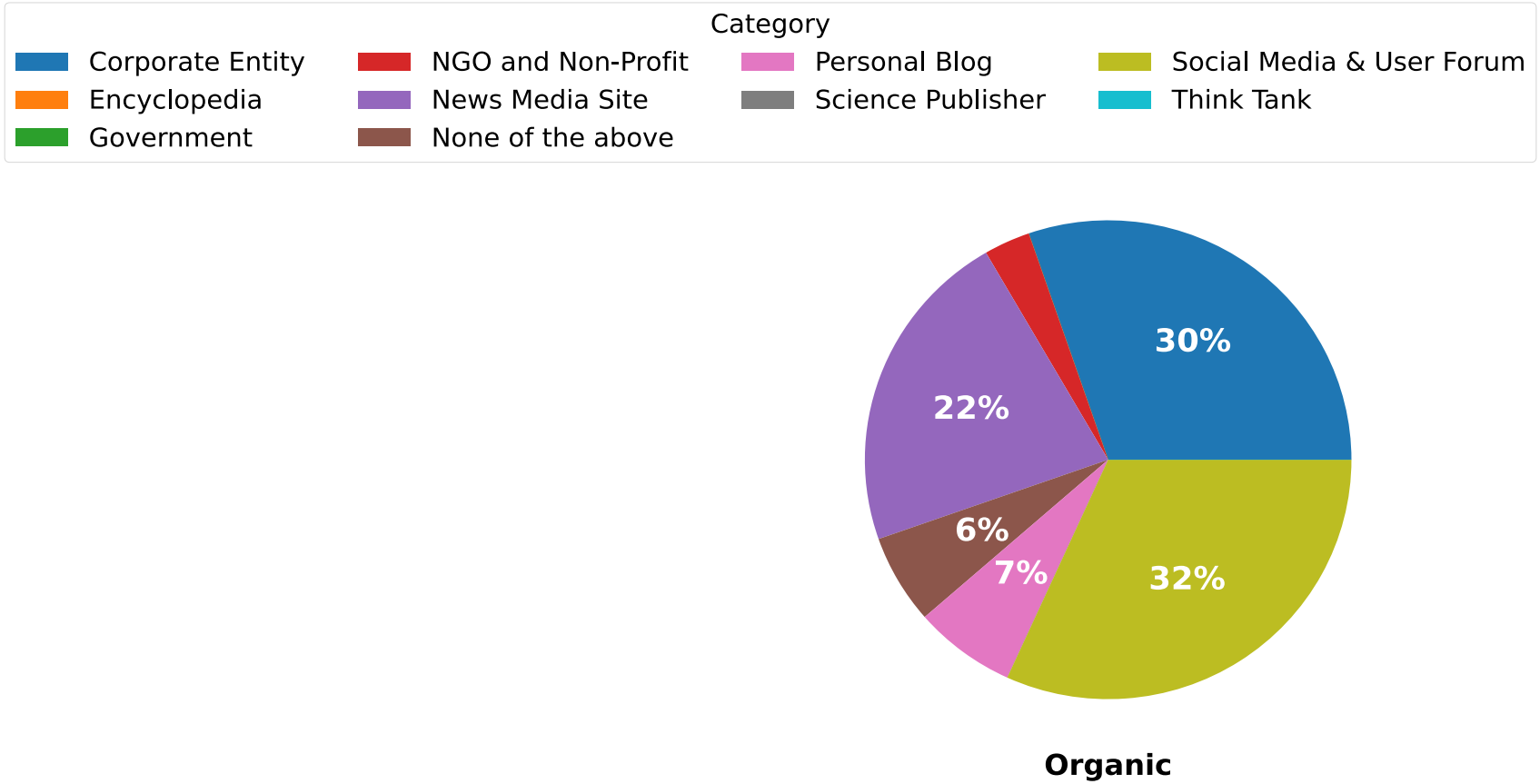
Which Sources Are Used?



AI Overview links found within the Top 100 Google links

Agents' sources are not overlapping much with traditional search

What Kinds of Sources Are Consulted?



Agents surface different information

Querying Search Agents Repeatedly

Query:

“Does the Soria province have a border with Navarre?”

Engine	Temp. = 0	
	$t_0 \rightarrow t_5$	$t_0 \rightarrow t_{24}$
GPT-Tool	9%	10%
GPT-Search	16%	17%
Gemini	15%	15%
AIO	17%	16%
Sonar	27%	28%

% Queries with a decision flip

LLM stochasticity can lead to undesirable effects

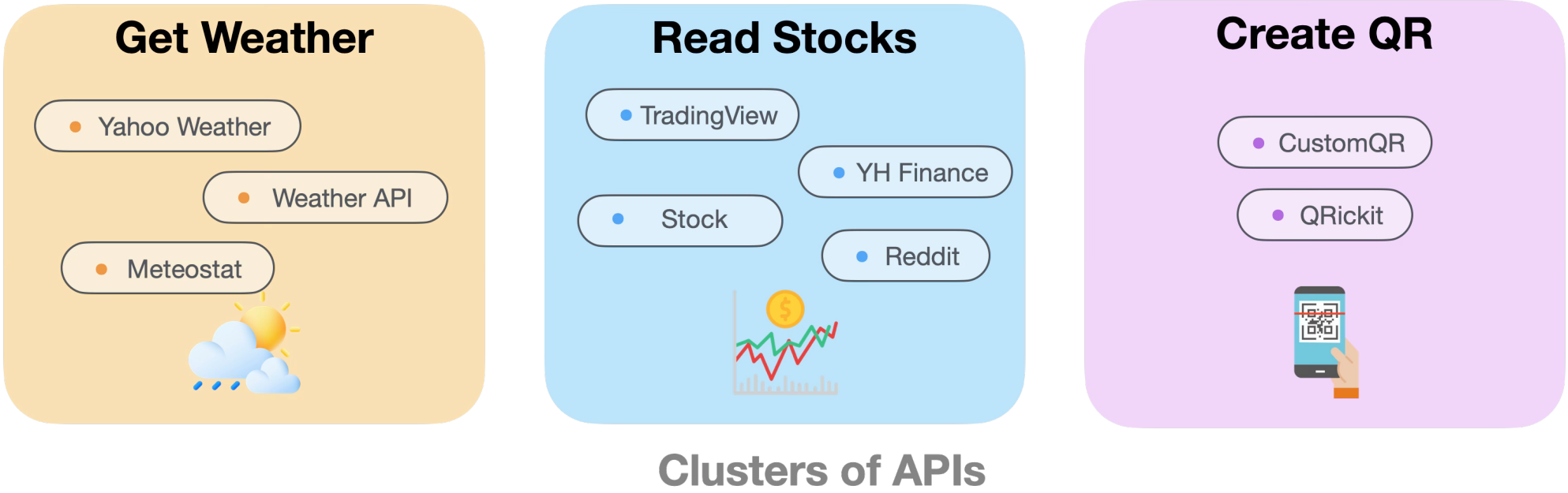
From Search Agents to Tool-Using Agents

What is the total cost of 3 nights at €137 per night plus 19% VAT, split between 4 people?

```
Thought: Need exact arithmetic.
Action: calculator("(3 * 137 * 1.19) / 4")
Observation: 122.3775
Final: Each person pays about €122.38.
```

When should the model delegate to an external system?

Millions of Tools Available



Clusters of APIs

- Marketplaces like RapidAPI, MCP servers, etc.
- Fast-growing ecosystems

Millions of Tools Available

	Example	Why it matters
Information tools	search, RAG, database lookup	give the model external knowledge
Computation tools	calculator, Python, test runner	give exact or executable feedback
Action tools	email, browser, calendar, file edits	let the agent change the world

Kinds of Tools

- **MCP (Model Context Protocol):**
  one standard so any model can call any tool

A tool call is a typed action

- Function calling:
  - the model emits a structured call the system executes
- The tool contract:
  - name, description, schema, errors, permissions
- The model chooses a policy over an action space it did not design
  - Failure modes: wrong tool, malformed arguments, hallucinated tools

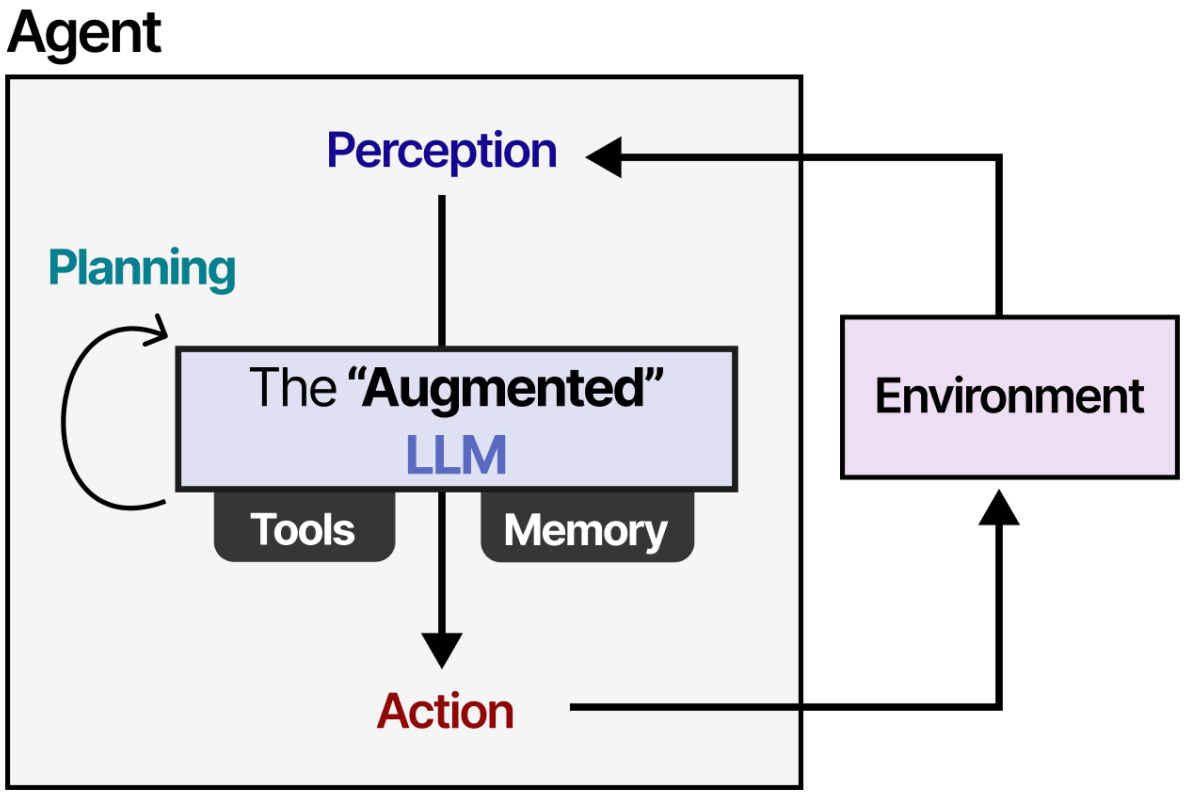
Tool Use ≠ Full Agency

- Tool-augmented LLM:
  - Question → Tool call → Answer

- Agent:
  - Goal → Decide → Act → Observe → Update state
    → Decide → Act → Observe → ...
    → Finish

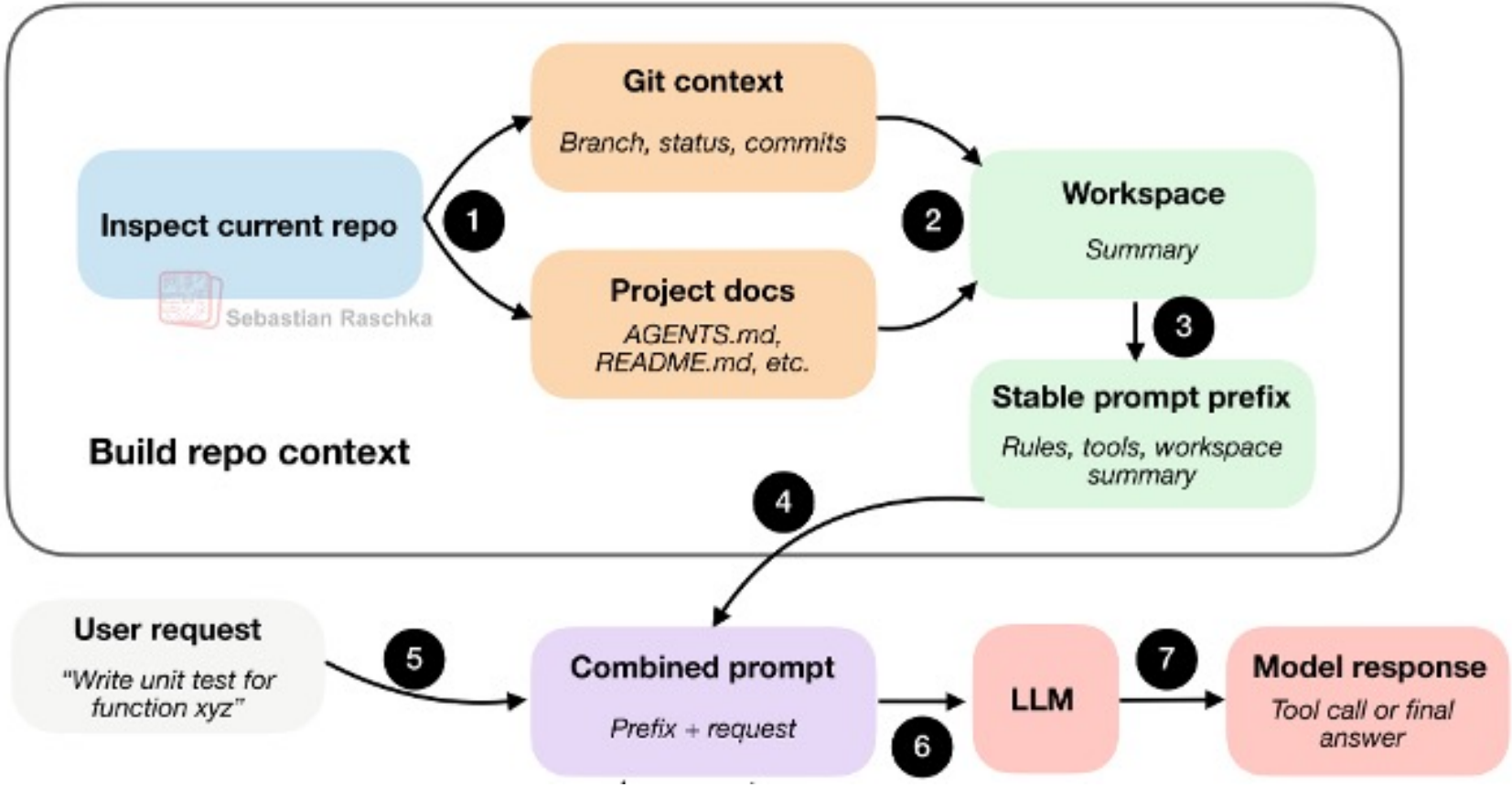
Agency requires a control loop

Agent Harness



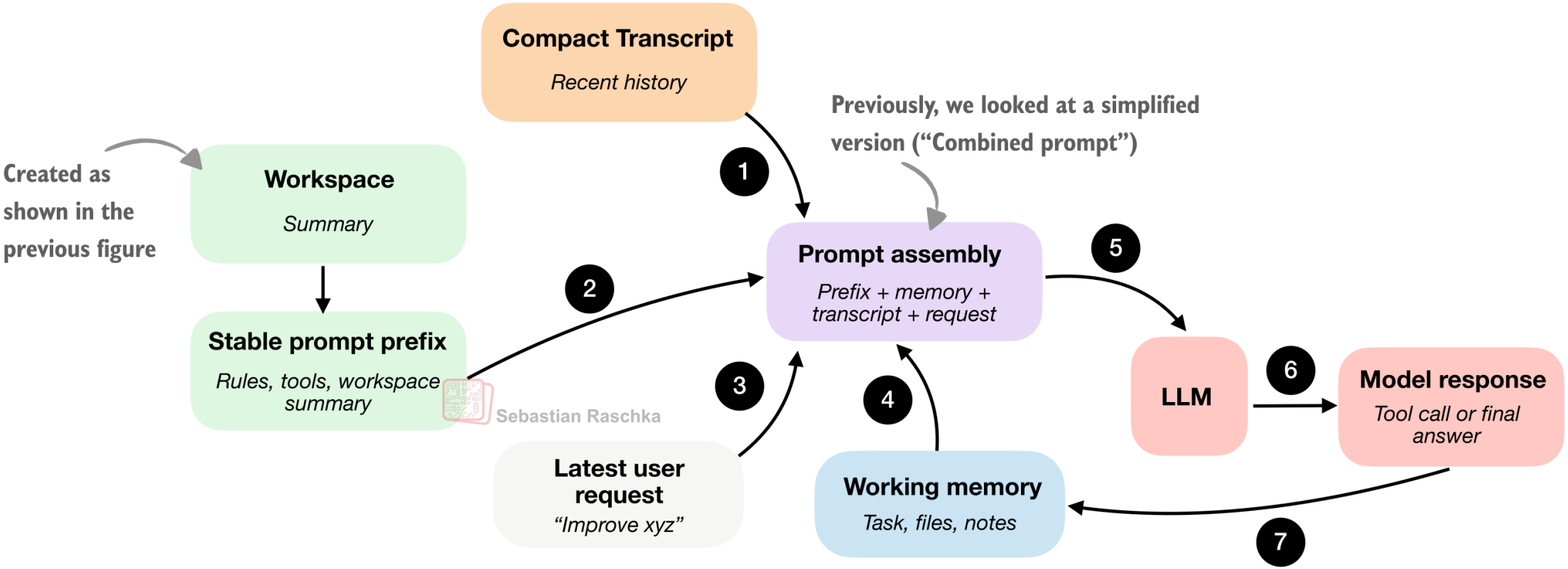
[A Visual Guide to LLM Agents]

Example: Context In Coding Agents



Blog Post: Components of A Coding Agent

Example: Context In Coding Agents



Blog Post: Components of A Coding Agent

How Do We Know Agents Are Good?

Benchmarks

- Web / OS / software: WebArena, OSWorld, SWE-bench
- Tool use:
  - τ-bench
  - economic work: GDPval
  - multi-agent: MultiAgentBench

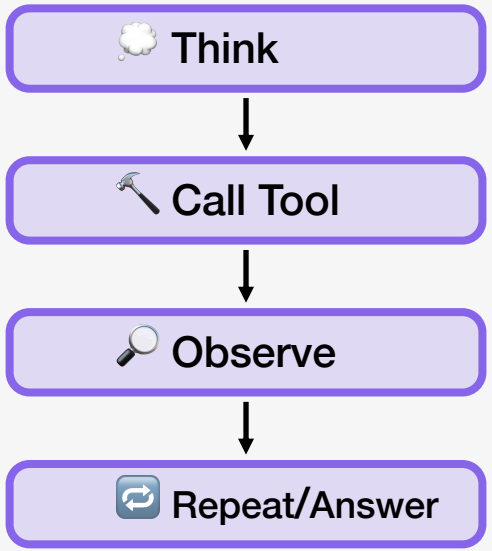
- Caveat for later: leaderboards report best-case capability

Exercise 1: Build an Agent (20 min)

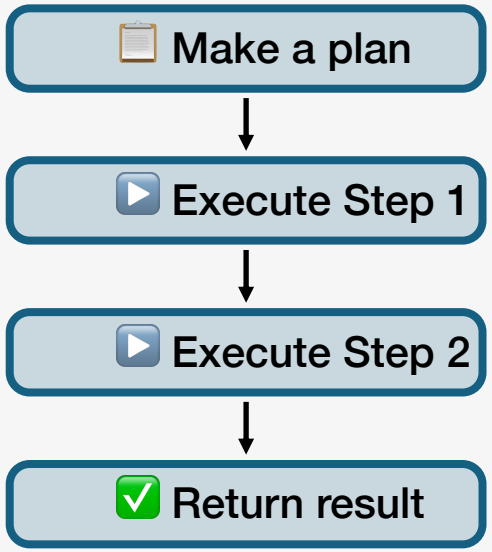
Part 2: Multi-Agent Systems

Common Agent Architectures

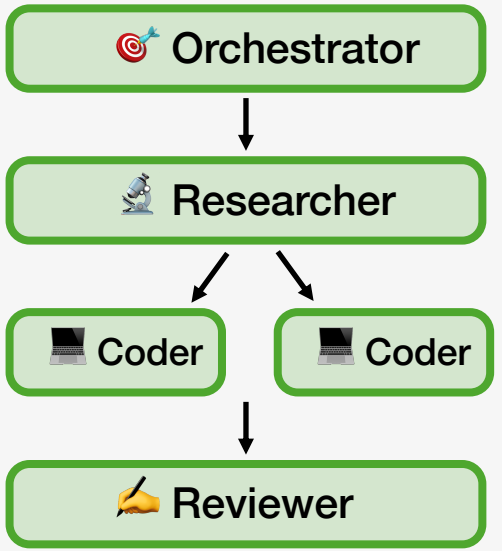
ReAct
Reason + Act Loop



Plan → Execute
Plan first, then act



Multi-Agent
Specialized roles



Why Move Beyond A Single Agent?

Single Agent

Planning,
Searching,
Executing,
Remebering,
Checking

Multi-Agent System

Capability

Decompose
Specialize
Explore

Efficiency

Parallelize
Route
Scale

Reliability

Verify
Redundancy
Recover

Control

Coordinate
Permissions
Audit

Multi-agent systems separate who plans, acts, knows, and shares.

Building Multi-Agent Systems

- Multiple agents, each with own tools, memory, planning
- Each subagent gets its own context window
- **Multi-agent ≠ multi-tool:** one agent calling many tools is still one agent
  - Caution: many “multi-agent” systems are really just multi-step pipelines

Workflows vs. Agents

[Anthropic: Building Effective Agents]

Workflow ⚙️

Predefined process

Developer specifies
- Which steps exist
- Their order and branches
- When the process stops

↓

Best when: task structure is known

Agent 🤖

Model controls process dynamically

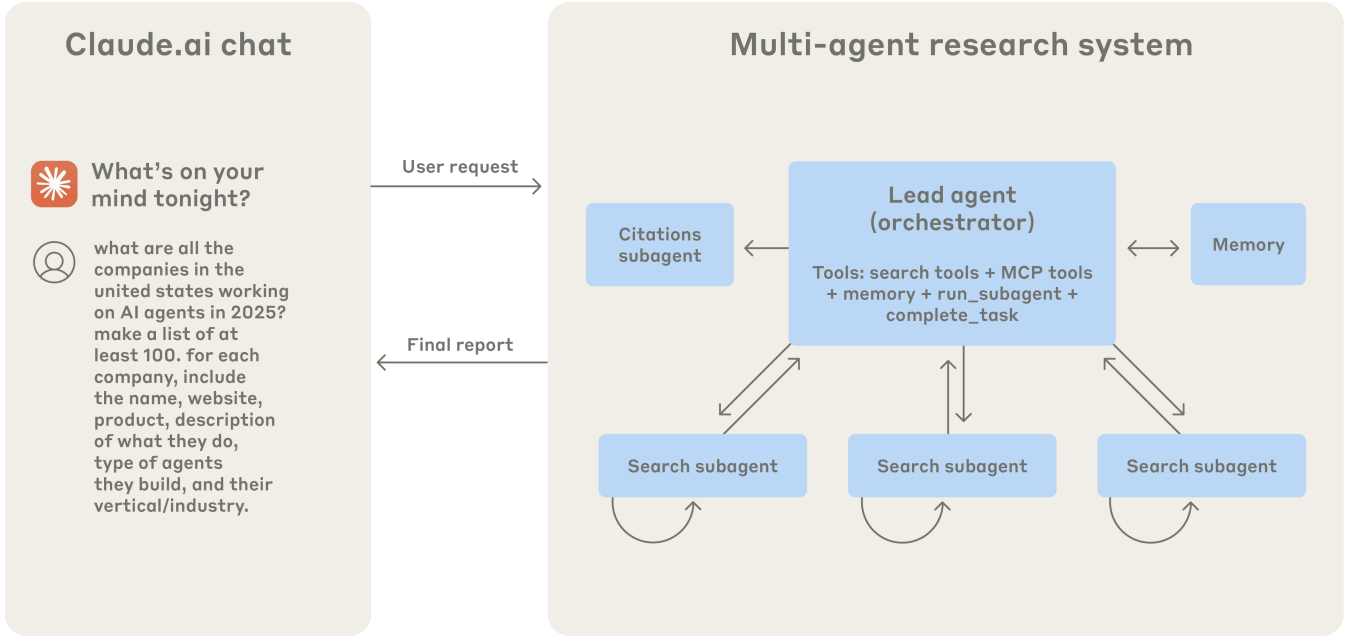
Model determines
- Which steps are needed
- Which tool to use next
- When to revise / stop

↓

Best when: path cannot be predicted

When Multi-Agent Helps

- Anthropic's orchestrator–worker system beat a single agent by ~90% on their eval
- Best for breadth-first work

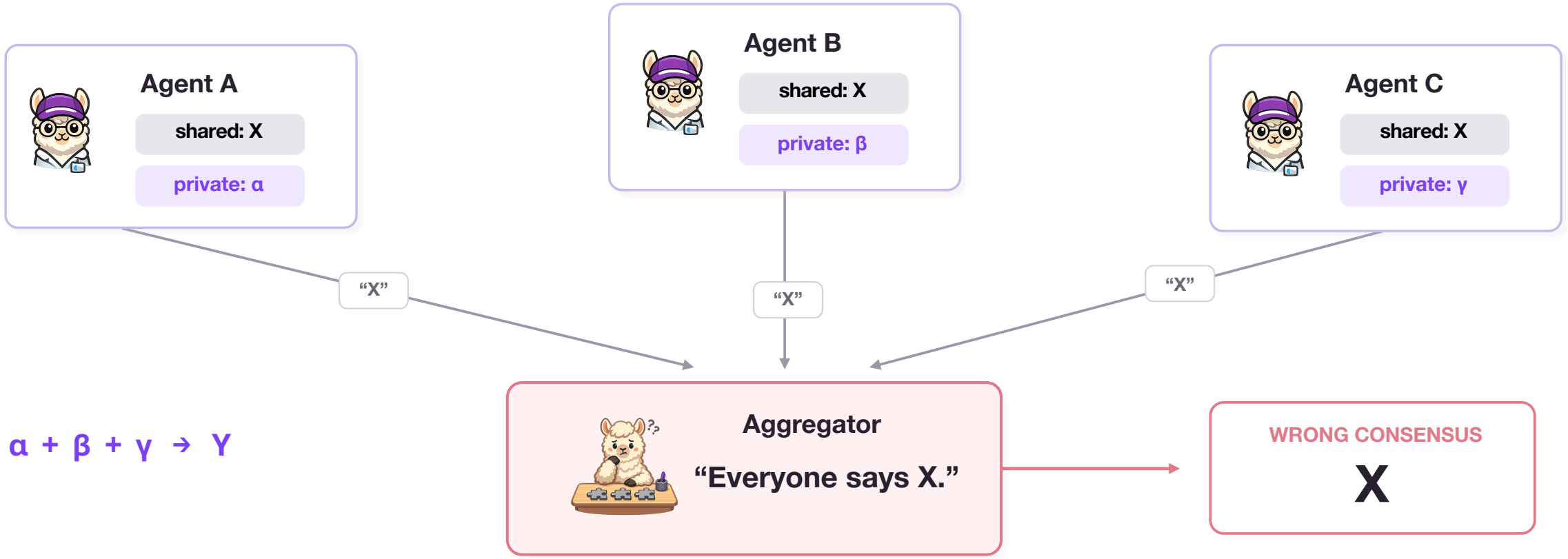


[Anthropic Multi-Agent Research System]

…But It Costs

- The same system uses ~15× the tokens of chat
- Token usage alone explained ~80% of the performance variance
- Every hop adds latency, cost, and a chance to lose information
- Less compelling when a single strong agent + good context already handles it

When Agent Communication Breaks



Multi-agent discussion can amplify shared information while suppressing unique evidence.

Many Design Choices

Orchestration = the runtime control layer:

who acts, when, with what context, under which rules, and how to merge / stop

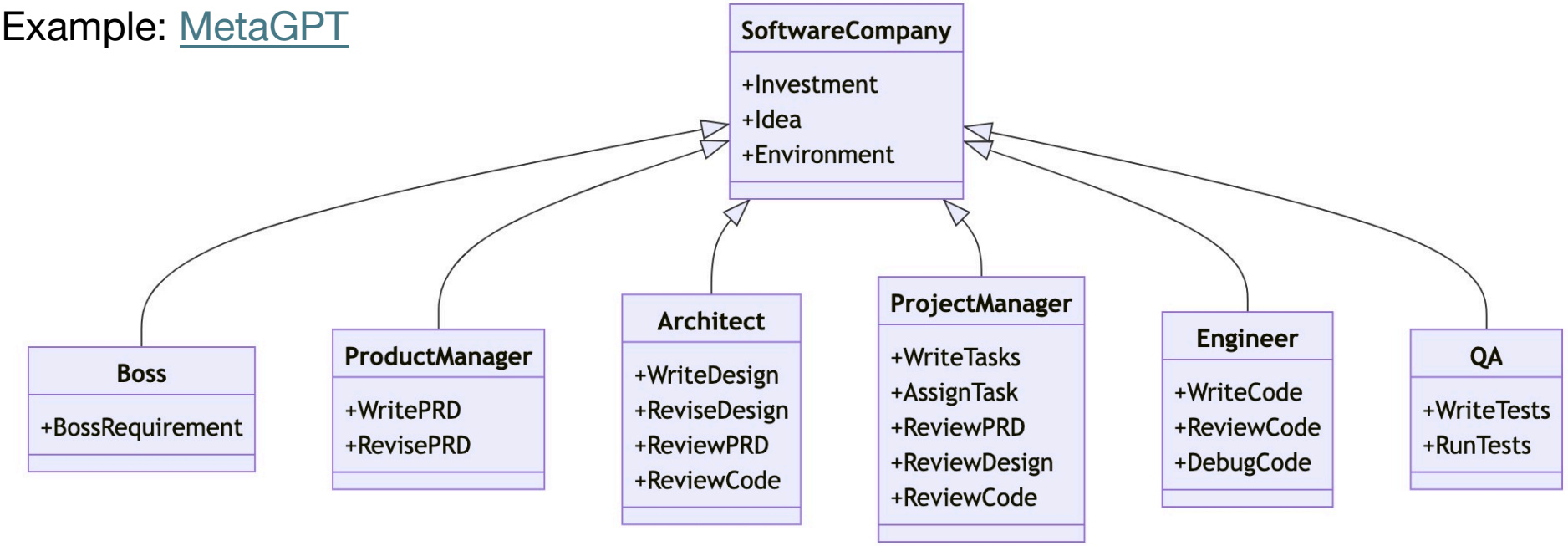
- **Design Choices:**
  1. *Role structure*
  2. *Topology*
  3. *Communication*
  4. *State / memory*
  5. *Termination*

Role Structure

Variants: homogeneous · role-specialized · dynamically generated

- Are all agents symmetric, fixed specialists, or spun up per task?

Example: MetaGPT



Six Recurring Roles

Worker / Specialist

Supervisor / Manager

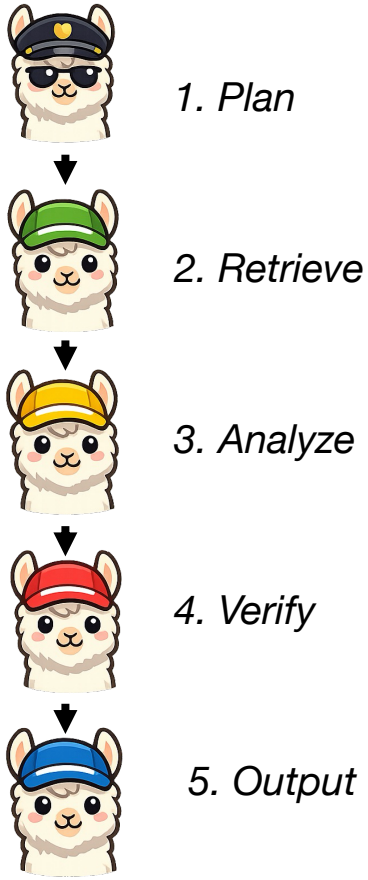
Router

Critic / Judge / Verifier

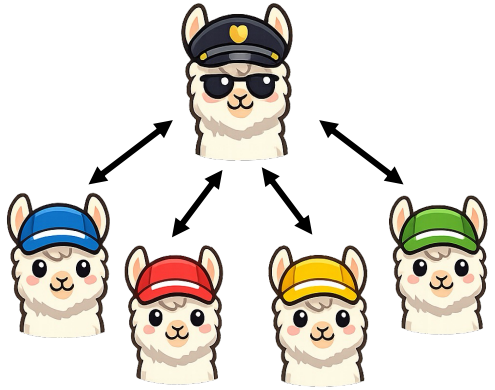
Memory / Retrieval Agent

Human Approval / Policy Gate

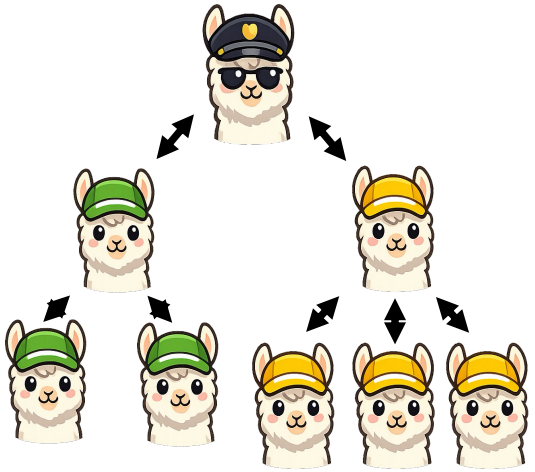
Who Talks To Whom



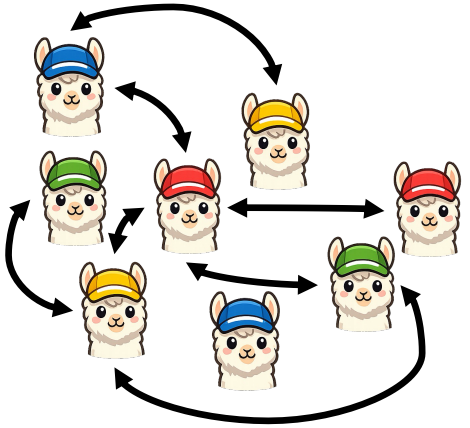
Chain/Pipeline: linear hand-off



Star: centralized supervisor orchestrates



Tree: hierarchical decomposition



Decentralized Peers

+ Hybrid Approaches

How To Communicate

- **Coordination medium**
  - NL messages
  - Structured state
  - Shared memory

- **Key factors:** cost, ambiguity, traceability, efficiency of inter-agent exchange

- **Coordination mode**
  - Sequential pipeline
  - Concurrent fan-out
  - Debate / group chat
  - Protocol driven collaboration

When Are We Done

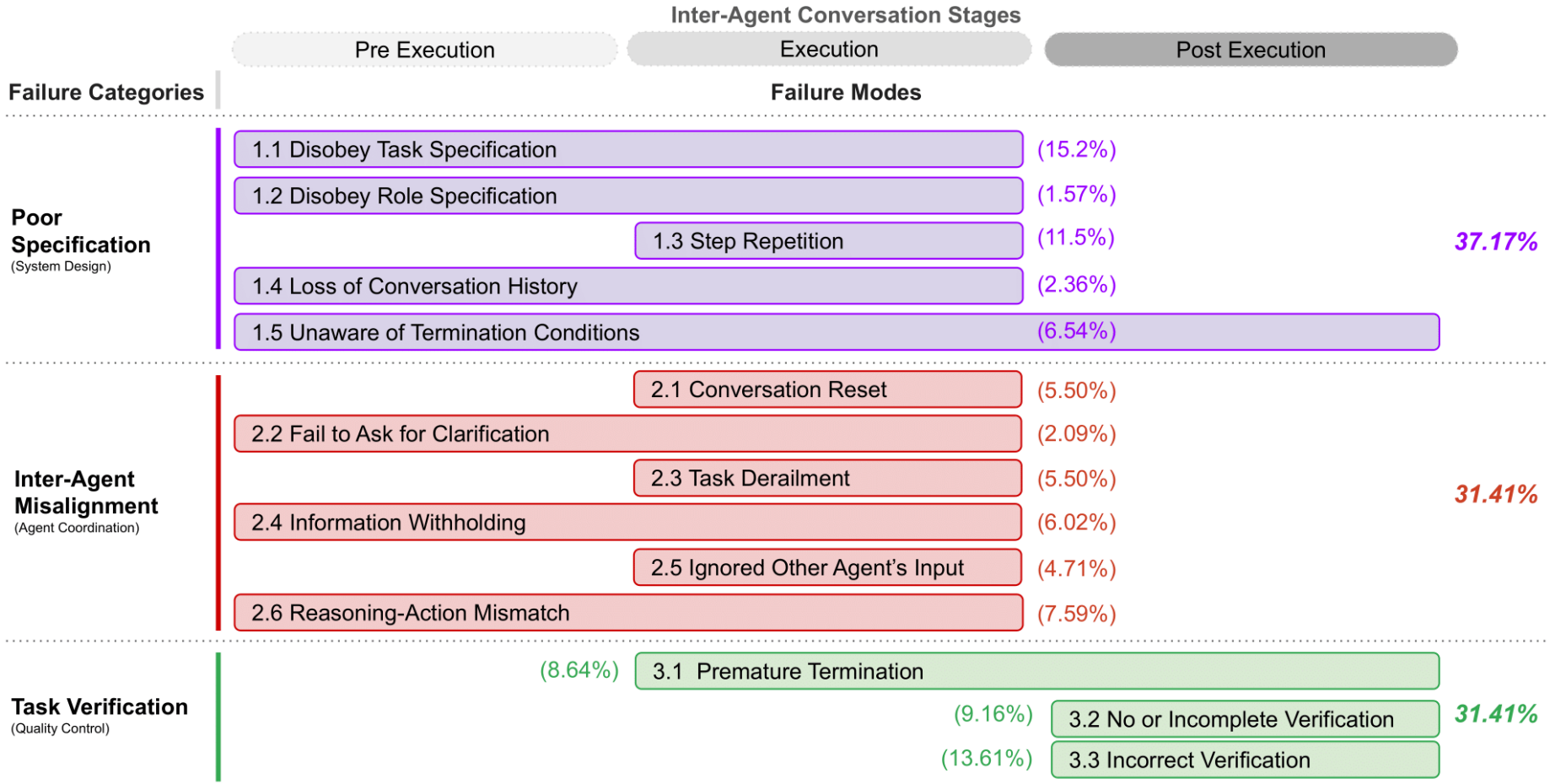
How to make decisions or reach consensus?

- Fixed depth
- Judge-based stop
- Approval-interrupt
- Adaptive stop

- **Also: different behavioral regimes**
  - Cooperation, competition, coopetition, self-play
  - Whether agents seek consensus, explore alternatives, or strategically oppose one another

Why Orchestration Breaks

- Multi-Agent System Failure Taxonomy (MAST)



More Agents ≠ Better

- Teams can underperform their best expert — even when told who it is (Pappu et al., 2026)
- Biased consensus: teams average views instead of weighting expertise (Okawa, 2026)
- Cost & latency multiply
  - Gains usually come from the orchestrator, not the agent count (Zywot et al., 2026)
  - The best system is the simplest one that reliably works

Exercise 2: Multi-agent orchestration (20 min)

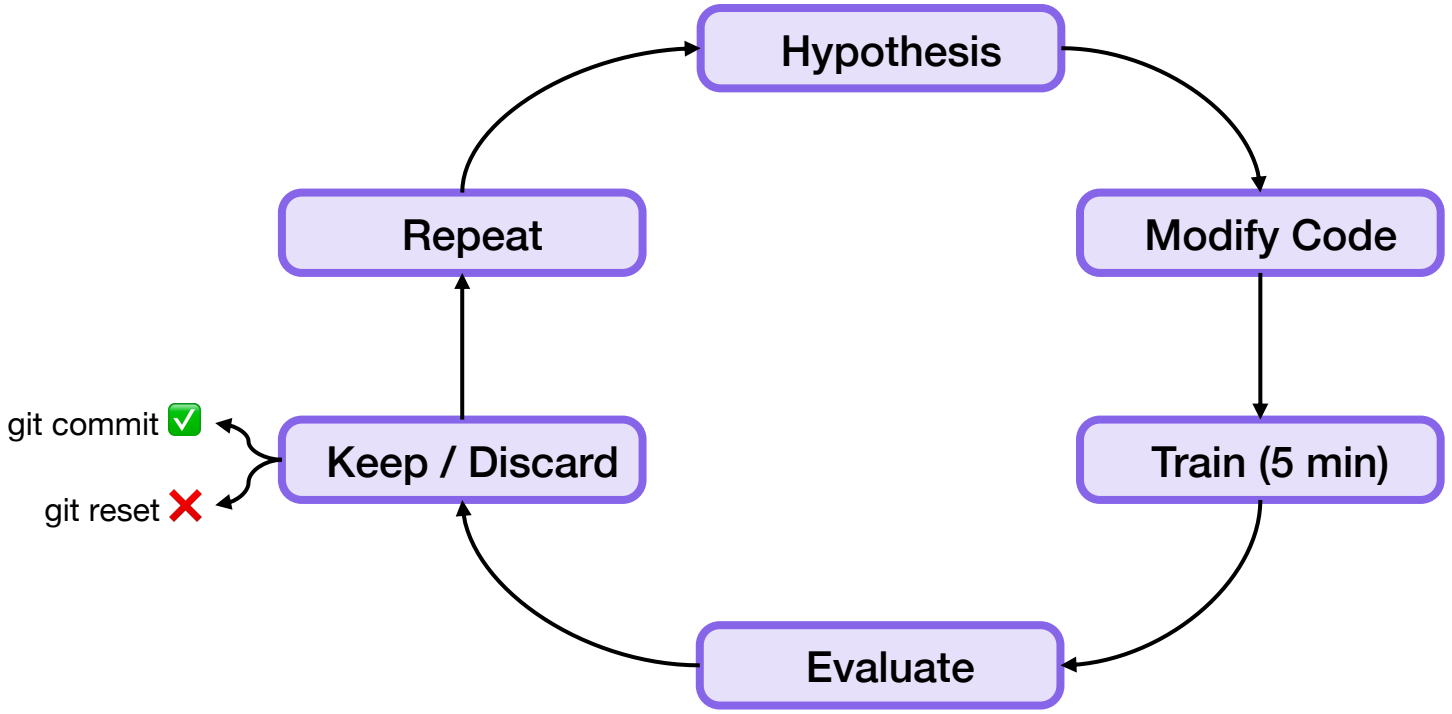
Outlook: Agentic AI for Science

From Tools to Co-Scientists

- Agents that generate hypotheses, propose experiments, read results, refine
- Reshaping what “doing research” means

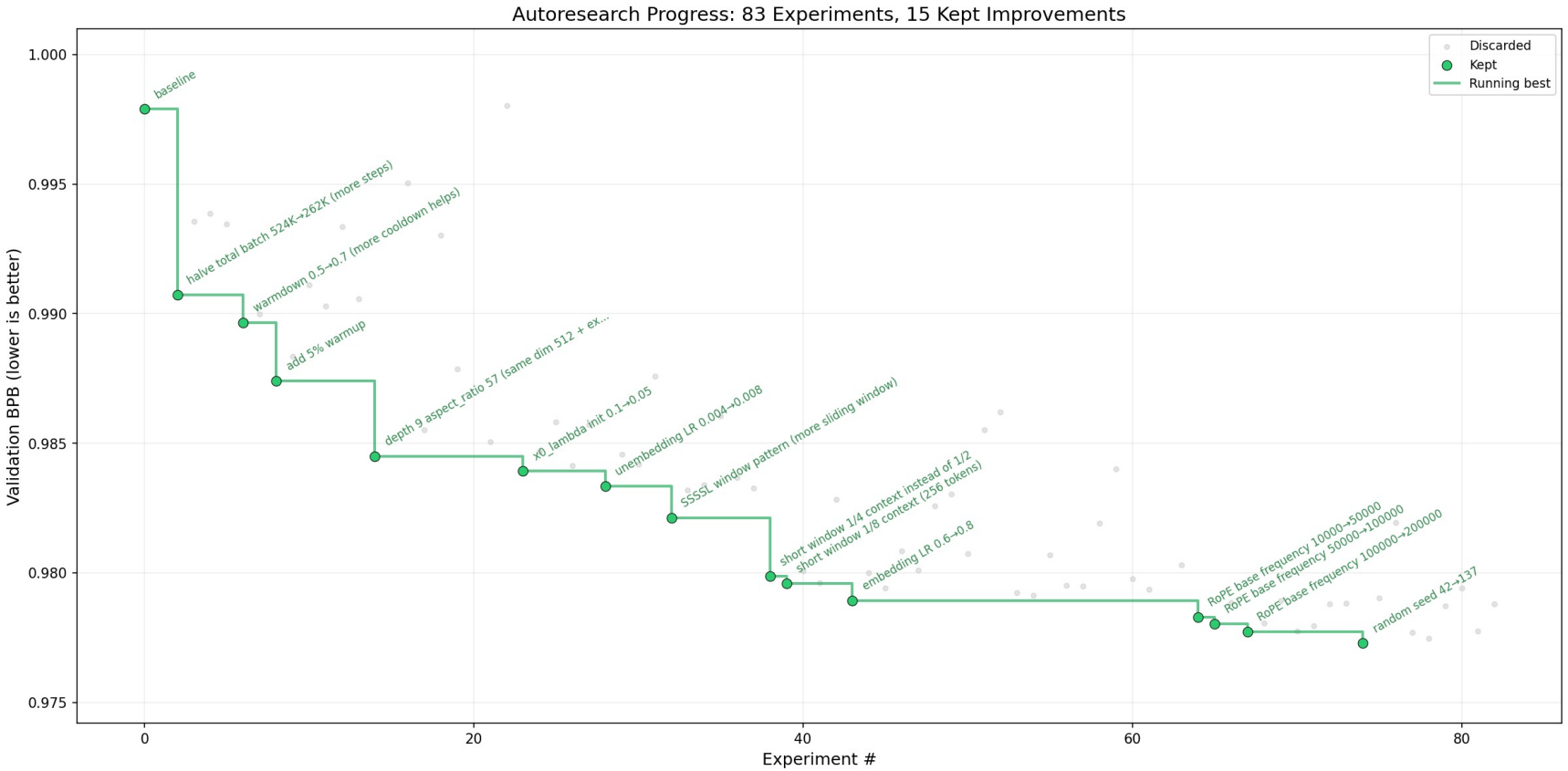
AutoResearch

- An agent loops overnight on one GPU
  - 2-day run: 700 experiments; “time-to-GPT-2” 2.02h → 1.80h



AutoResearch

- An agent loops overnight on one GPU
  - 2-day run: 700 experiments; “time-to-GPT-2” 2.02h → 1.80h



Examples of Agentic AI for Science

- **Biomedical Research Agents**
  - DeepMind Co-Scientist (Nature 2026)
  - FutureHouse Robin
  - Stanford Biomni
  - OriGene
  - Google Co-Scientist
- **General Scientific Research Platforms**
  - Google Gemini for Science
  - Microsoft Discovery
  - Anthropic Claude Science
- **Autonomous Computational Research Systems**
  - Sakana AI: AI Scientist v2
  - Karpathy's Autoresearch
  - Google ERA / AlphaEvolve

The Reality Check

- **Reproducibility:** discovery trajectories are stochastic, prompt-sensitive
- **Narrowed exploration:** agents recombine near existing ideas, rarely leap
  - (Tang et al., 2026)
- **Evaluation is the moat:** human-level with trusted eval
  - (Rabanser et al., 2026)

Key Takeaways

- **Agents are becoming more default** — e.g., already in web search
- **An agent is a loop with tools, planning, and memory**
- **Orchestration distributes control; more agents ≠ always better**
- **Agentic science vastly impacts the researcher's job**

Questions?

We are hiring!

Thank You!

🔗 elisabethkirsten.com

✉️ elisabeth.kirsten@rub.de

🏢 informatik.rub.de/aisoc

References

- Our Lecture “LLM Engineering”: github.com/aisoc-lab/llm-engineering-lecture

- Read more:
  - Grootendorst, *A Visual Guide to LLM Agents* (2025)
  - Raschka, *Components of a Coding Agent* (2026)
  - Cemri et al., *Why Do Multi-Agent LLM Systems Fail?*
  - Anthropic, *How we built our multi-agent research system*
  - Anthropic, *Building Effective Agents*
  - *Accelerating scientific discovery with Co-Scientist*, Nature 2026
  - Kirsten et al., *Characterizing Web Search in the Age of Generative AI*
  - Wu et al., *To Call or Not to Call*
  - Dash et al., *The Algorithmic Self-Portrait*