

Snorkel

Data-centric Principles for AI Engineering



Vincent Sunn Chen

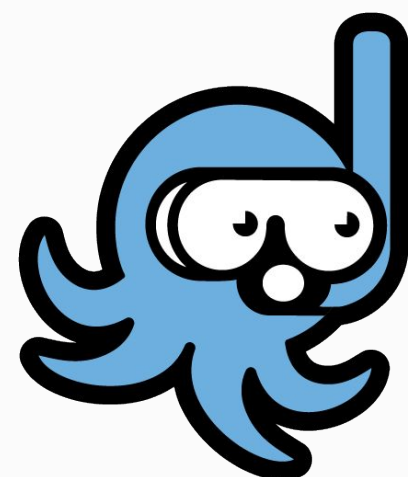
Head of ML Engineering // Founding Engineer @ Snorkel AI
@vincentsunnchen

Vincent Sunn Chen

Founding Engineer, Leading ML Engineering @ Snorkel

Previously, Researcher @ Stanford AI Lab

@vincentsunnchen





Debuggability + iteration
are critical to AI engineering.



Debuggability + iteration
are enabled by **data-centric**
development.

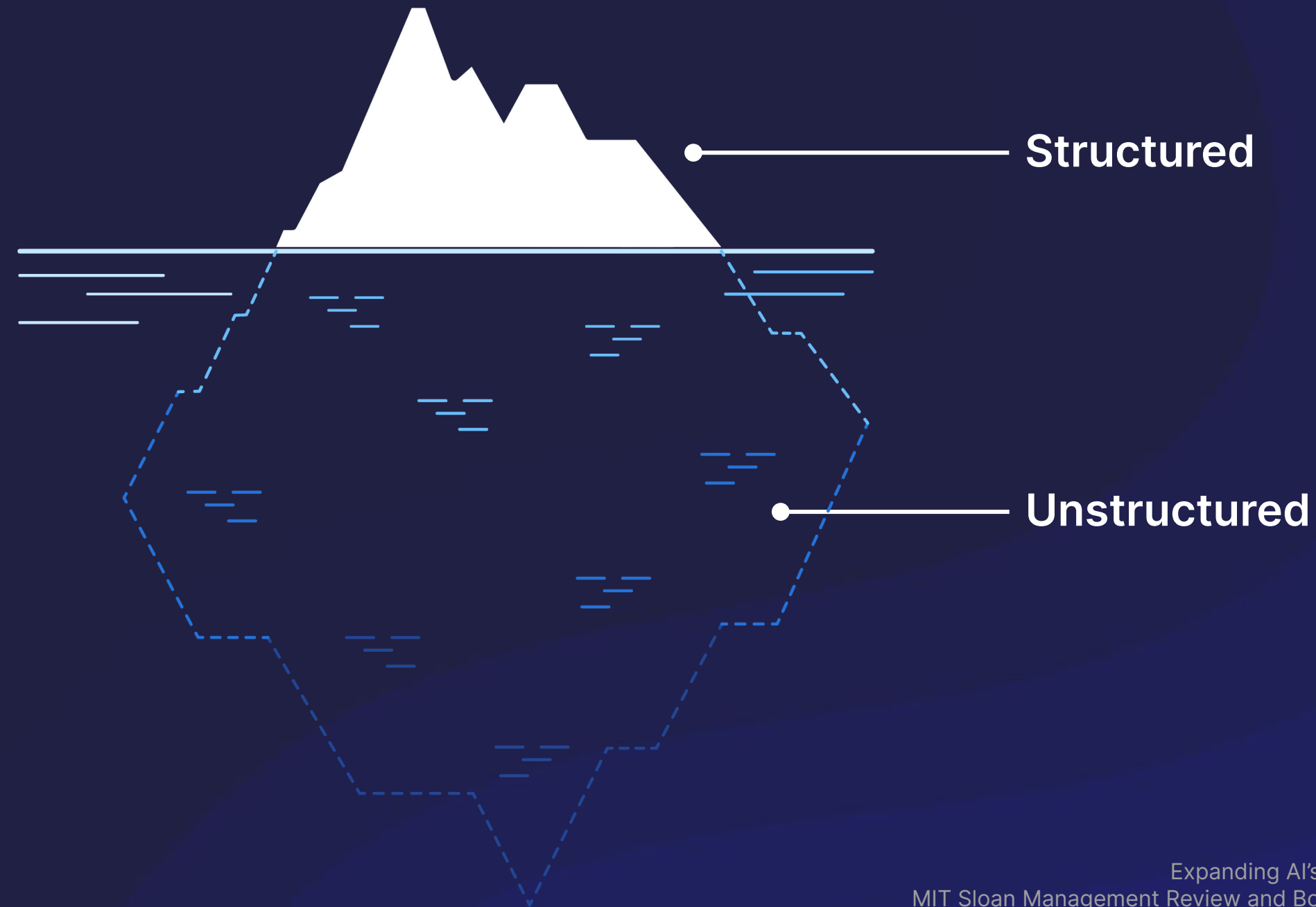
Outline

- Today's AI Engineering Challenges
- Data-centric Principles
- Case Study: Social Media Monitoring

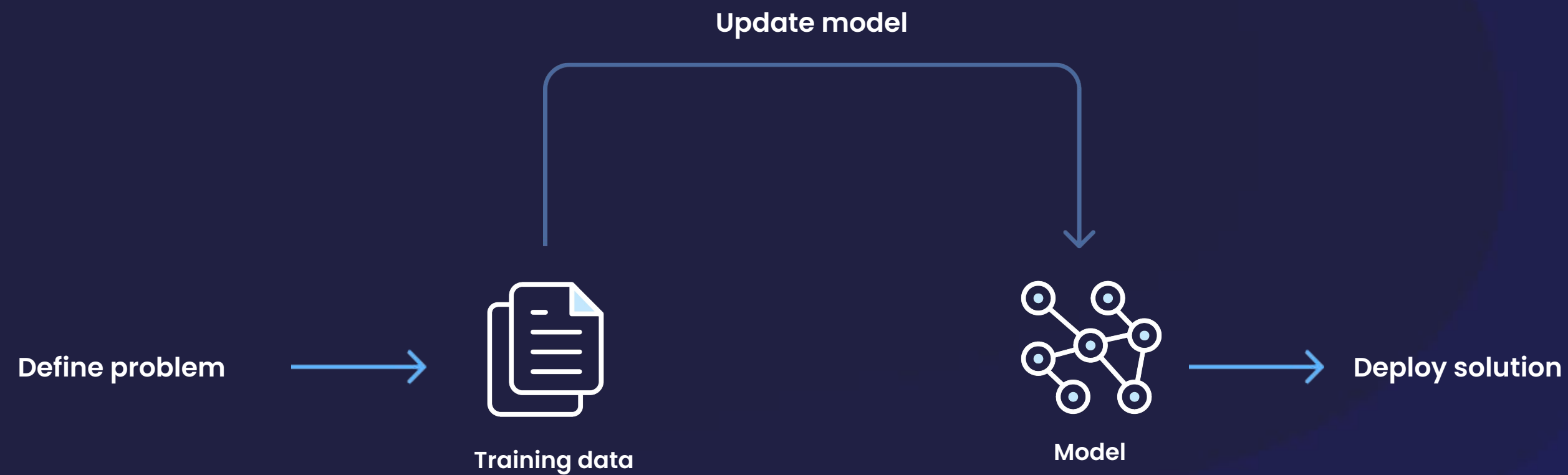
Today's AI Engineering Challenges



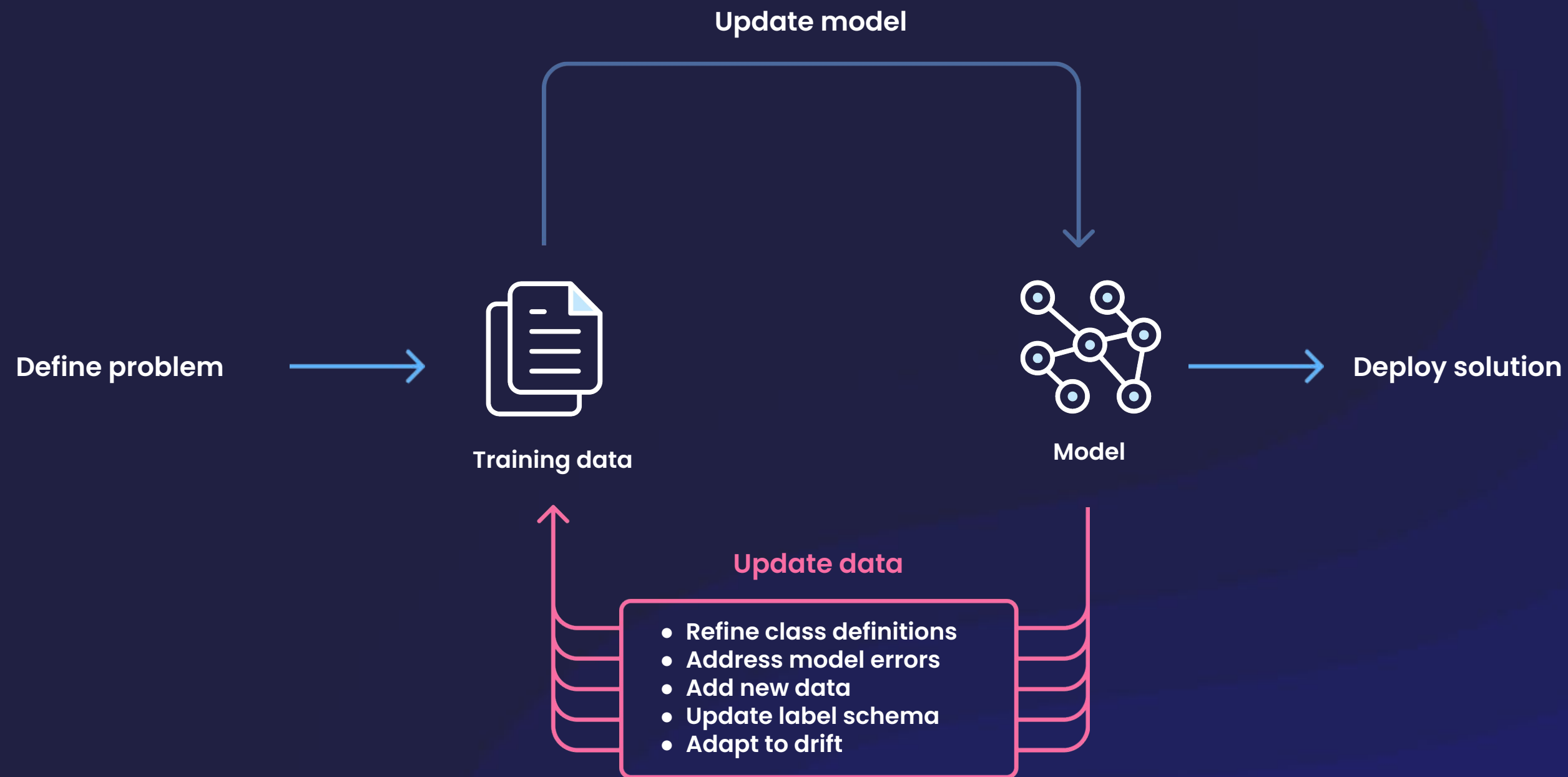
85% of organizational data is *unstructured, unlabeled, and not ready for AI use*



Training data development is iterative— not a one-time process



Training data development is iterative— not a one-time process

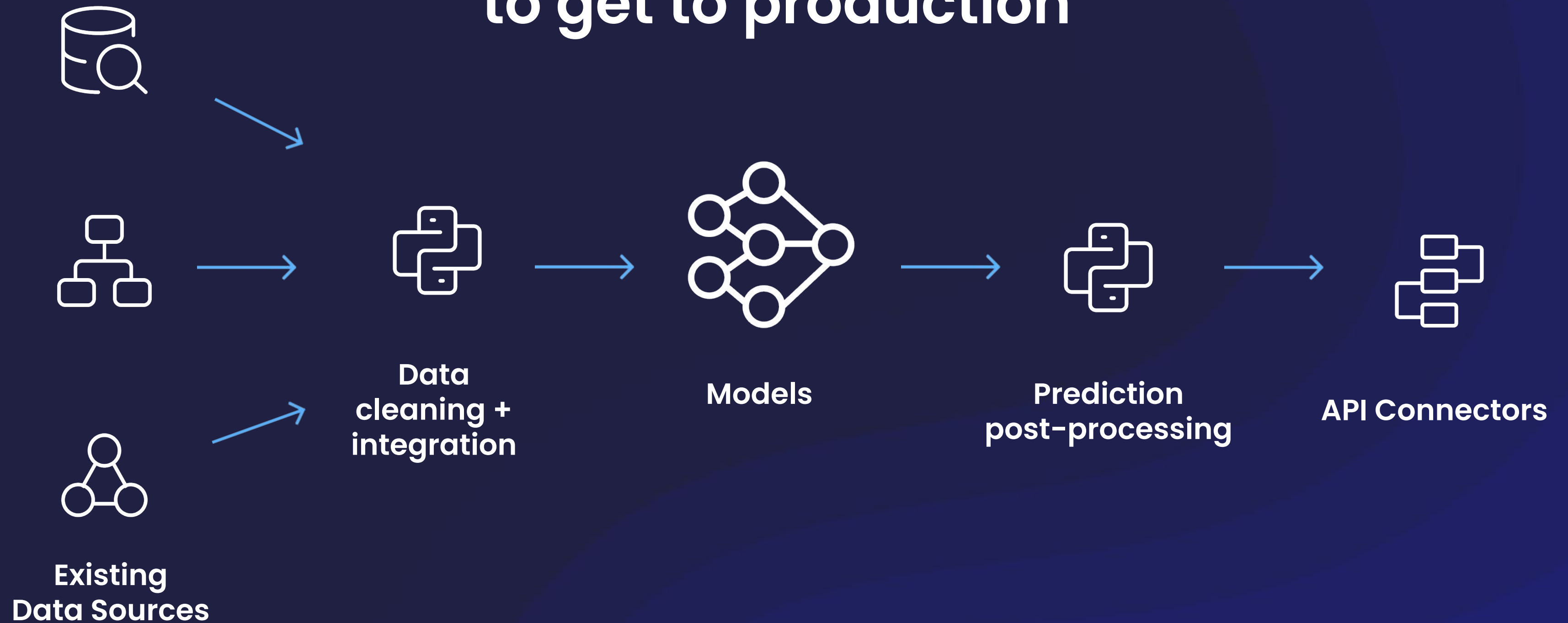


AI applications are about more than the model



Models

AI applications require data operations to get to production





Soumith Chintala  @soumithchintala · Jul 18 ...

Deep Learning is not yet enough to be the singular solution to most real-world automation. You need significant prior-injection, post-processing and other engineering in addition.

Hence, companies selling DL models as an API have slowly turned into consulting shops.

 29





 236

 1.4K







AI engineering is shifting from
model-centric to **data-centric**

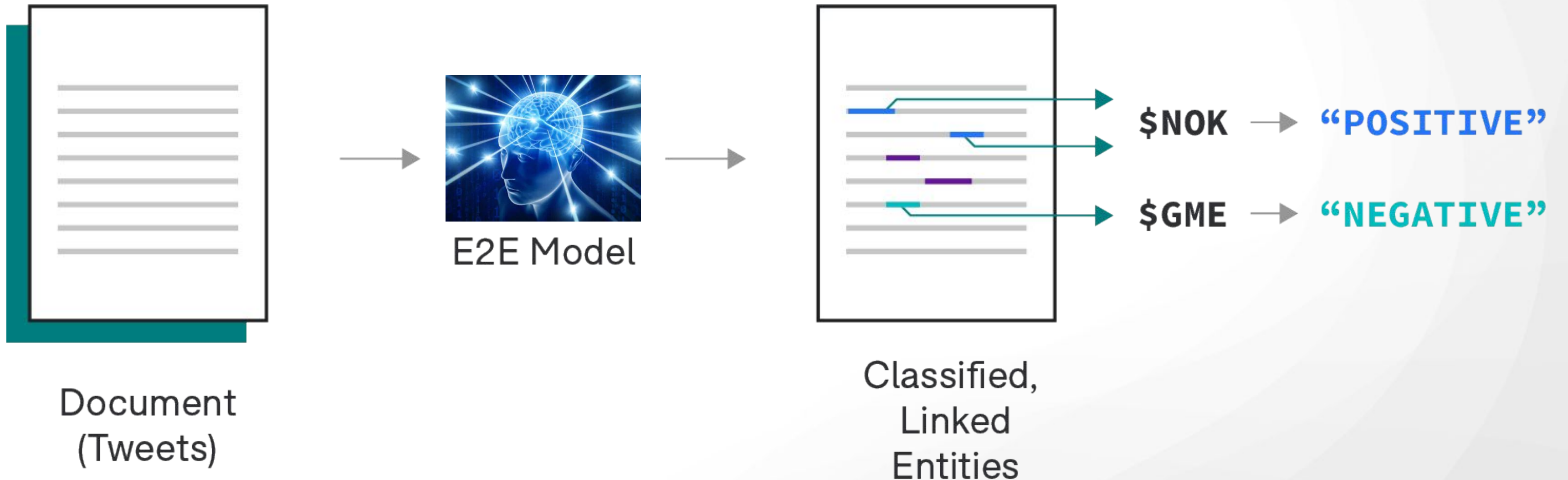
Data-centric principles for AI engineering

-  **Down with the end-to-end mega model**
-  **Long live end-to-end (evaluation and iteration)**
-  **ML should not be the universal default**
-  **Rapidly iterate with programmatic labeling**

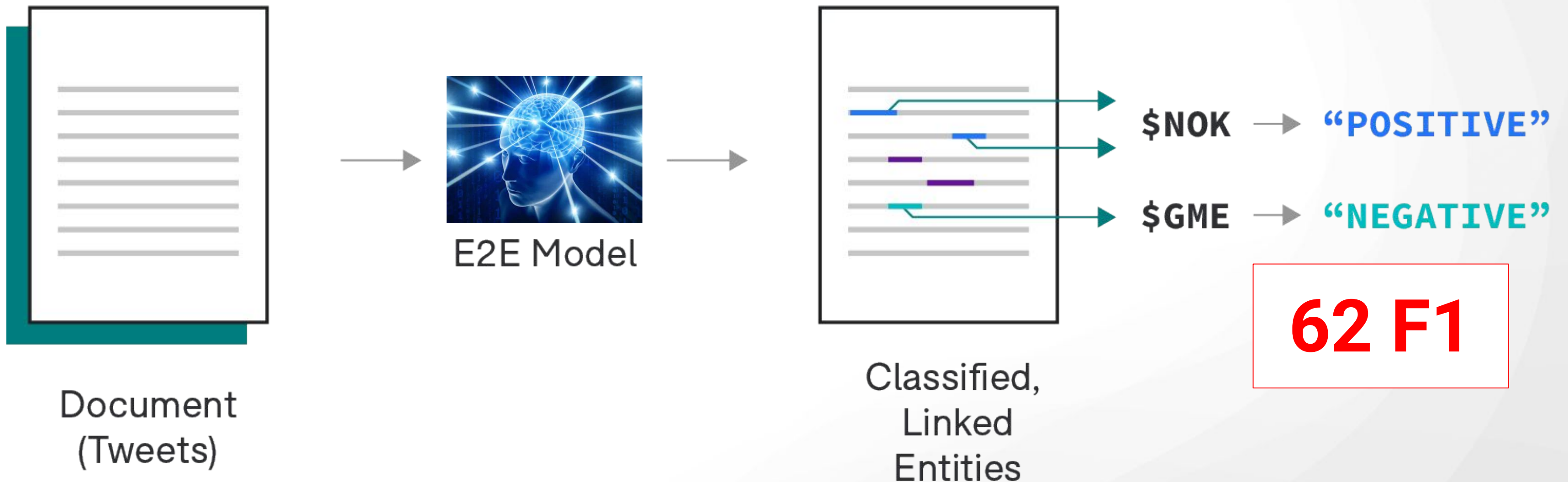
Data-centric principles for AI engineering

-  **Down with the end-to-end mega model!**
-  Long live end-to-end (evaluation and iteration)
-  ML should not be the universal default
-  Rapidly iterate with programmatic labeling

Down with the end-to-end mega model!

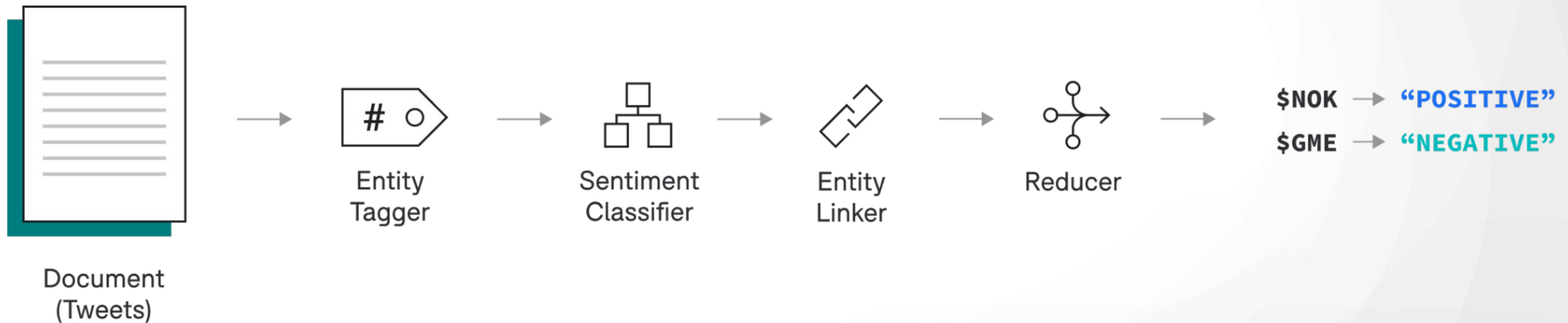


Where are the mistakes coming from?







Debuggability + introspection is a must-have!

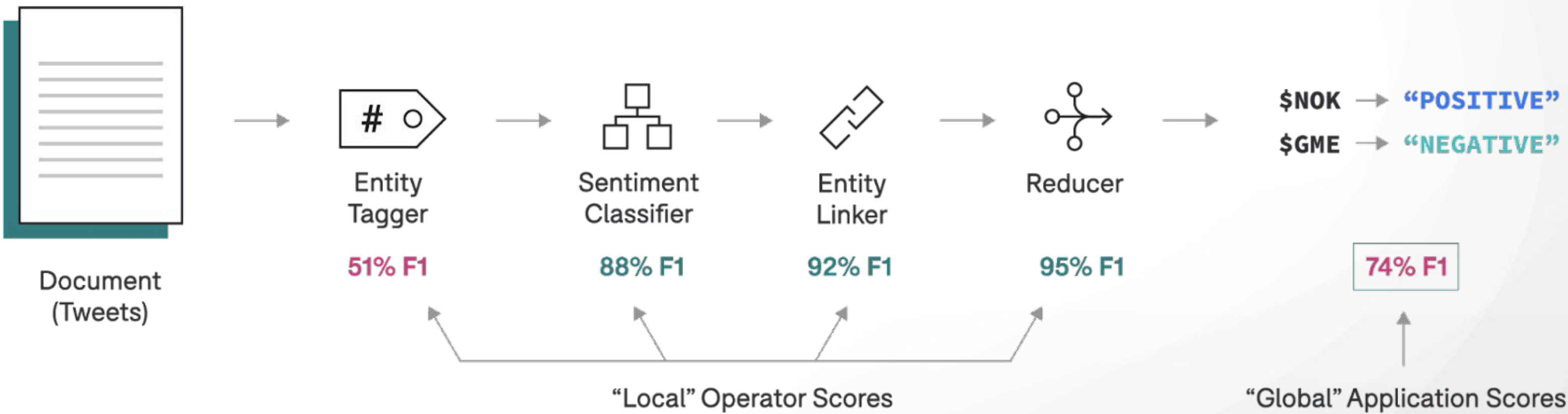
Decompose complex models into a pipeline of modular, debuggable building blocks.







Data-centric principles for AI engineering

-  Down with the end-to-end mega model!
-  **Long live end-to-end (evaluation and iteration)**
-  ML should not be the universal default
-  Rapidly iterate with programmatic labeling

Enable *local* (per-component) and *global* (end-to-end) evaluation and iteration.



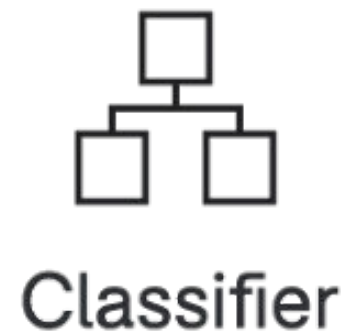
Data-centric principles for AI engineering

-  Down with the end-to-end mega model!
-  Long live end-to-end (evaluation and iteration)
-  **ML should not be the universal default**
-  Rapidly iterate with programmatic labeling

Each building block performs a *dataframe transformation*...

Subject	Body	Timestamp
Free money	...	Feb 02, 2021

Document DataFrame



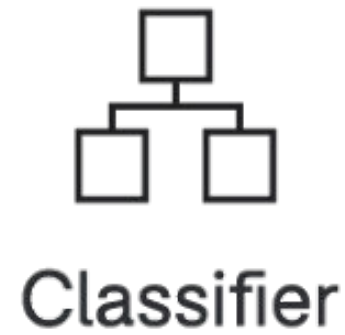
Subject	Body	Timestamp	Prediction
Free money	...	Feb 02, 2021	SPAM

Classified Document DataFrame

Building blocks are modular!

Subject	Body	Timestamp
Free money	...	Feb 02, 2021

Document DataFrame



Subject	Body	Timestamp	Prediction
Free money	...	Feb 02, 2021	SPAM

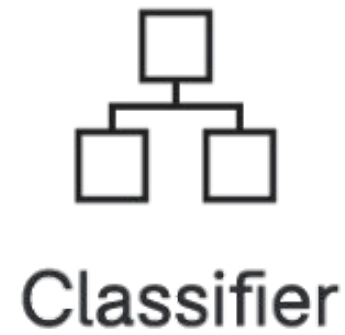
Classified Document DataFrame

Heuristic Classifier: "free money" in subject → SPAM

Building blocks are modular!

Subject	Body	Timestamp
Free money	...	Feb 02, 2021

Document DataFrame



Subject	Body	Timestamp	Prediction
Free money	...	Feb 02, 2021	SPAM

Classified Document DataFrame

Learned Classifier:

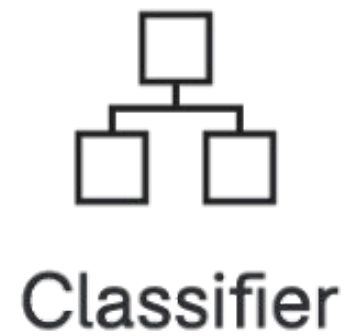


.predict(df)

Building blocks are modular!

Subject	Body	Timestamp
Free money	...	Feb 02, 2021

Document DataFrame







Subject	Body	Timestamp	Prediction
Free money	...	Feb 02, 2021	SPAM

Classified Document DataFrame

Start simple... then swap in ML as needed!

Data-centric principles for AI engineering

-  Down with the end-to-end mega model!
-  Long live end-to-end (evaluation and iteration)
-  ML should not be the universal default
-  **Rapidly iterate with programmatic labeling**

Iteration is bottlenecked by **manual data labeling**

Outsource to labeling vendors



- ⚠ Privacy challenges
- ❓ Lacks domain expertise
- 📦 Not auditable or governable
- 🔄 Hard to adapt

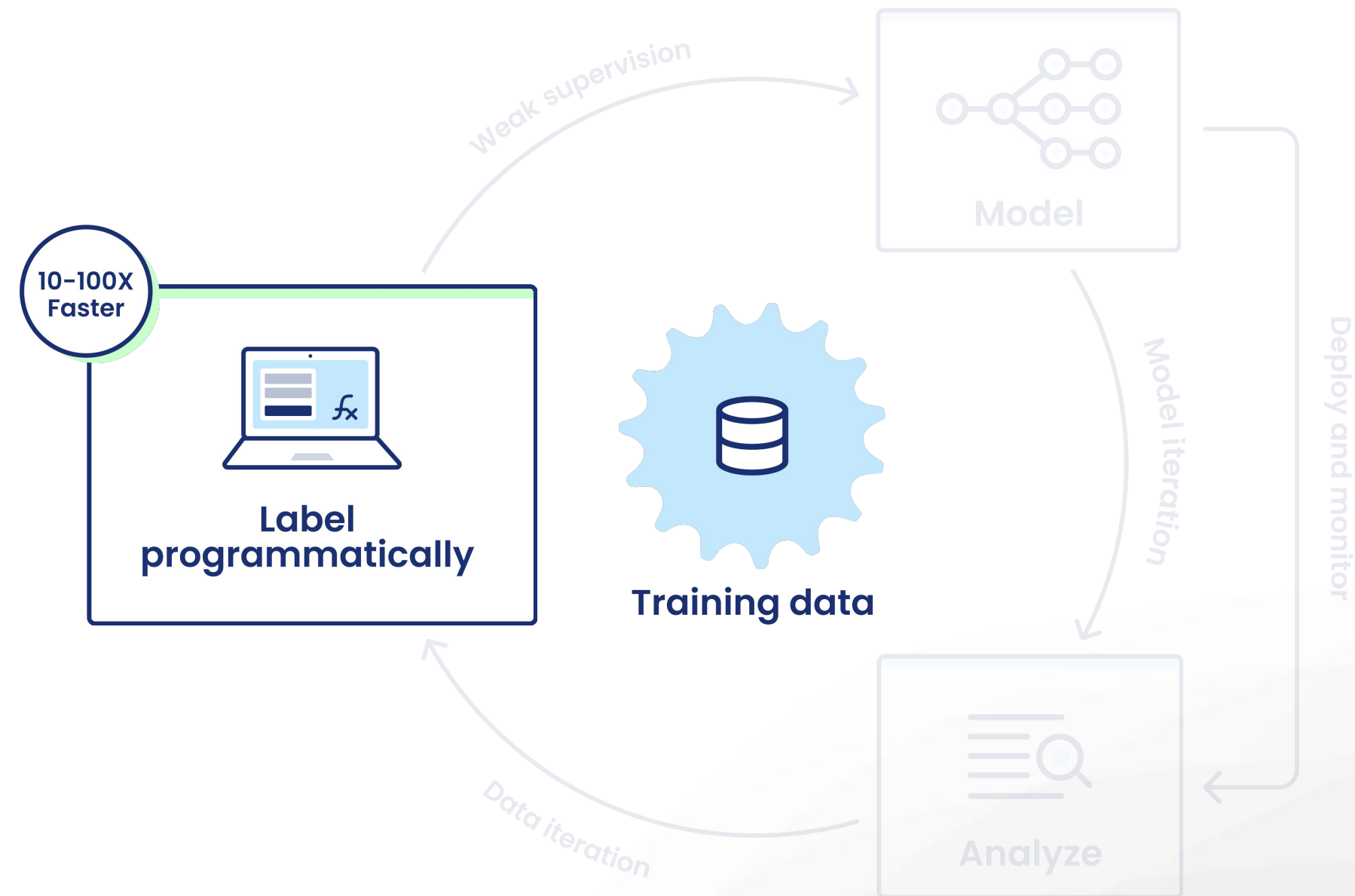
Label with in-house experts



+ Active learning, model assisted labeling, etc.

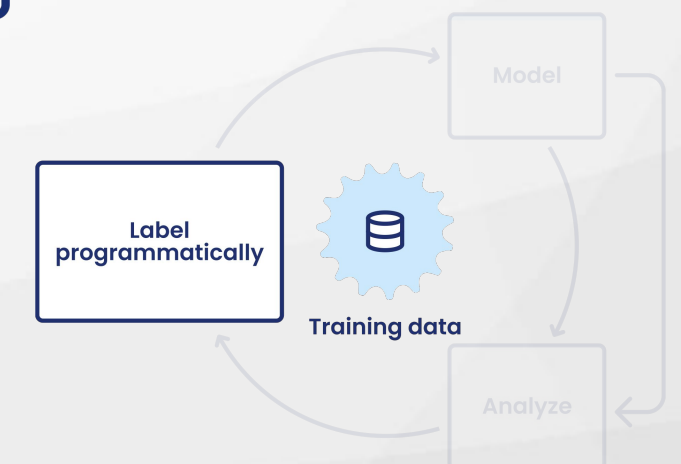
- 🕒 Slow
- 💰 High opportunity cost of domain experts
- 📦 Not auditable or governable
- 🔄 Hard to adapt

Create **labeling functions**, not manual labels

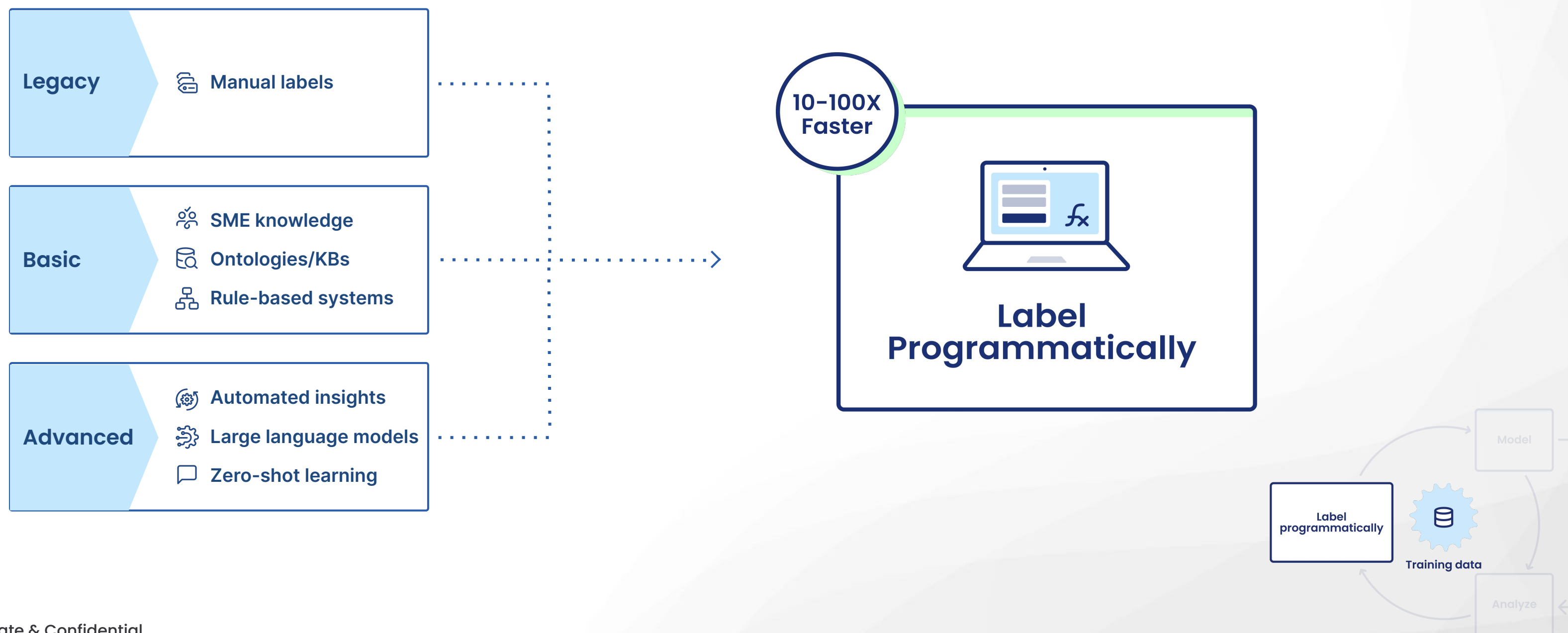


Create **labeling *functions***, not manual labels

```
"If "free money" is found  
in x.email..."
```



Snorkel Flow operationalizes organizational knowledge for AI



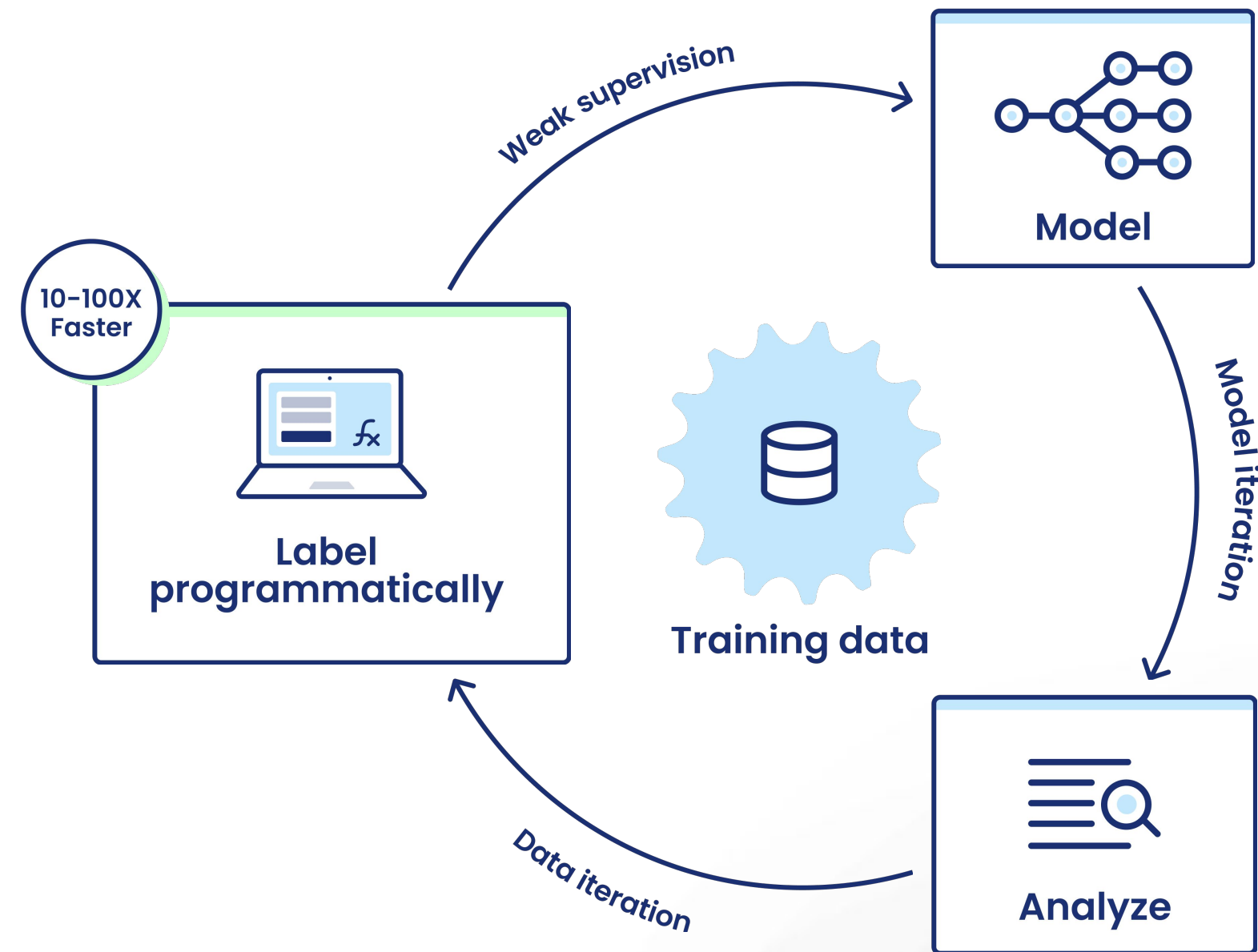
Rapidly build training datasets with labeling functions







Ratner et. al., NeurIPS'16;
Bach et. al., ICML'17;
Ratner et. al., VLDB'18;
Ratner et. al., AAI'19;
Varma et. al., ICML'19; etc.

<https://snorkel.ai/programmatic-labeling/>

Programmatic labeling enables **rapid iteration**, not complete manual relabeling



Data-centric principles for AI engineering

-  **Down with the end-to-end mega model**
-  **Long live end-to-end (evaluation and iteration)**
-  **ML should not be the universal default**
-  **Rapidly iterate with programmatic labeling**

Social Media Monitoring

The background of the slide features a series of overlapping, wavy, semi-transparent shapes in a color palette ranging from light pink and lavender to deep magenta and red. These shapes create a sense of movement and depth, primarily concentrated on the right side of the frame, while the left side remains a clean, white space.



Social Media Monitoring



Goal: Monitor public company sentiment on Twitter.



Document
(Tweets)



Classified,
Linked
Entities

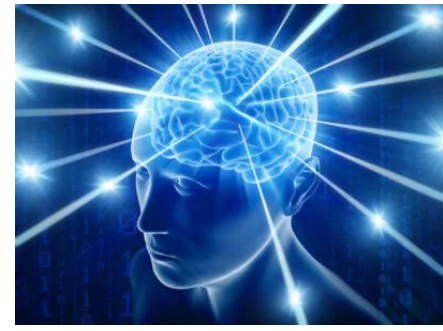
\$NOK → “POSITIVE”

\$GME → “NEGATIVE”

First pass...



Document
(Tweets)



E2E Model



Classified,
Linked
Entities

\$NOK → **“POSITIVE”**

\$GME → **“NEGATIVE”**

First pass...



Document
(Tweets)



E2E Model



Classified,
Linked
Entities



\$NOK



“POSITIVE”

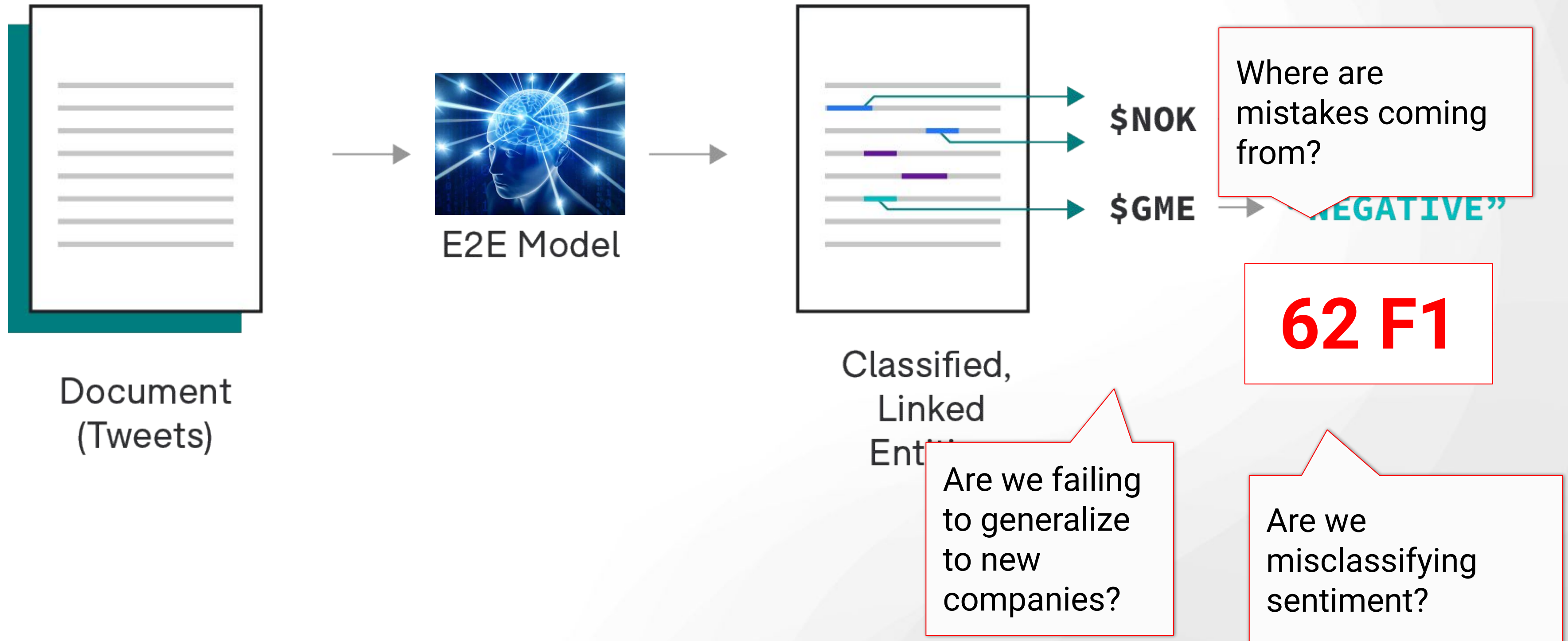
\$GME



“NEGATIVE”

62 F1

During MLE standup...



Document
(Tweets)

E2E Model

Classified,
Linked
Entities

\$NOK

\$GME

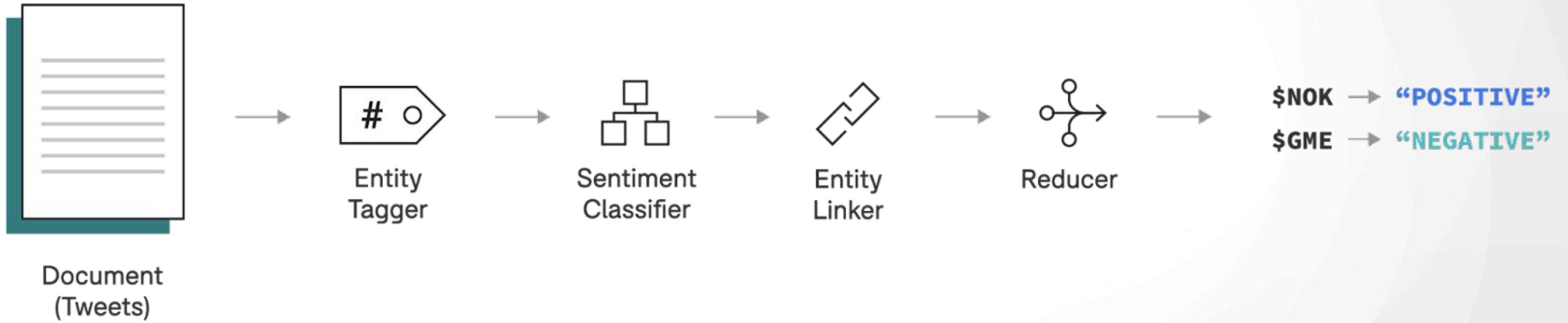
Where are
mistakes coming
from?

62 F1

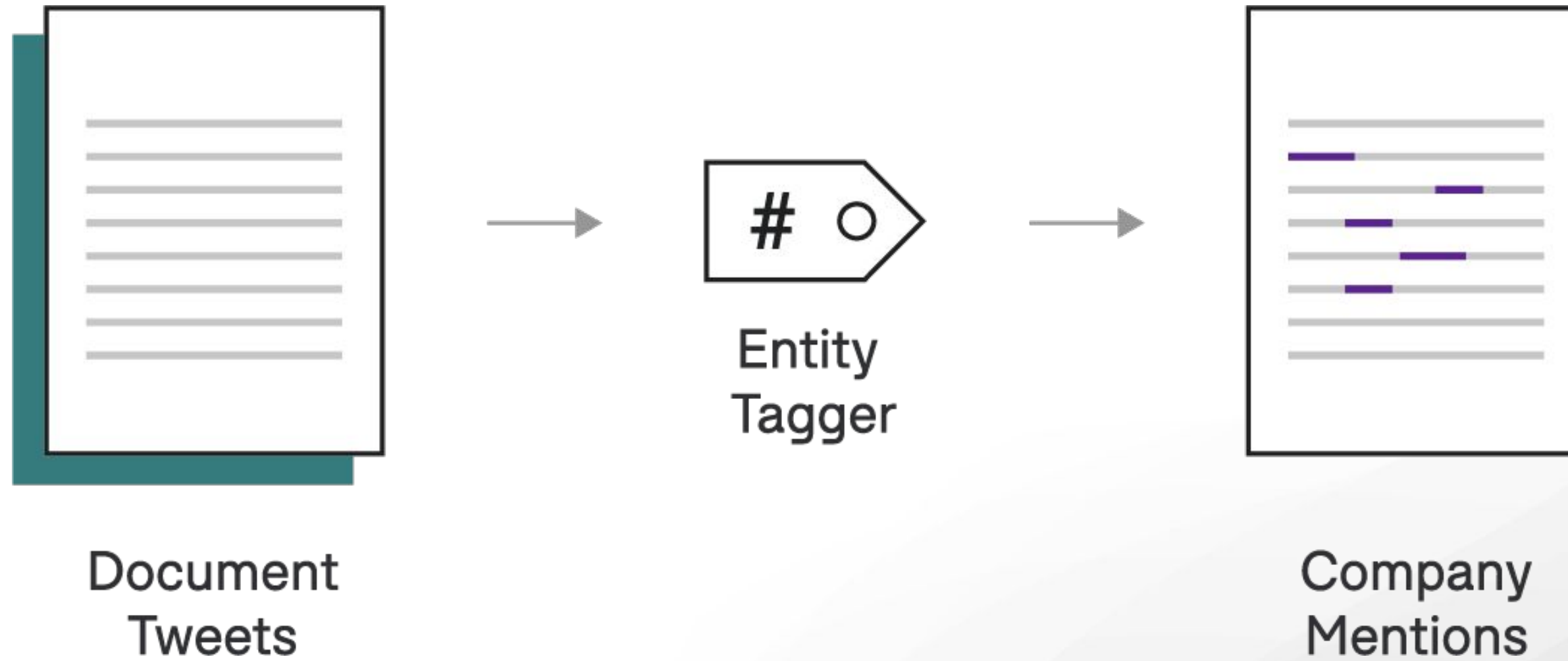
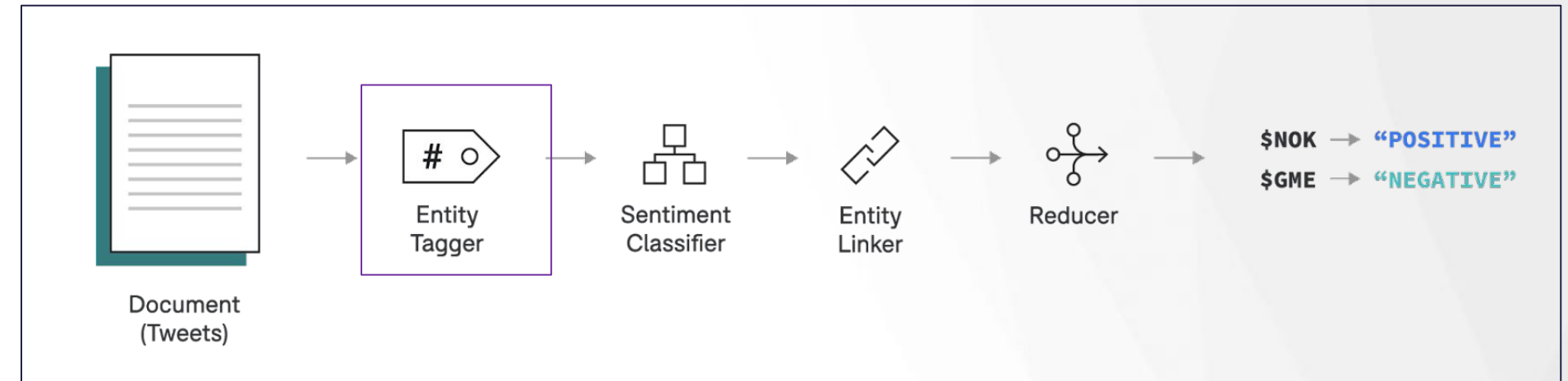
Are we failing
to generalize
to new
companies?

Are we
misclassifying
sentiment?

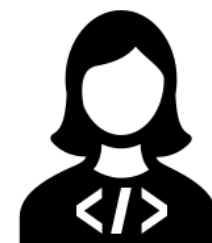
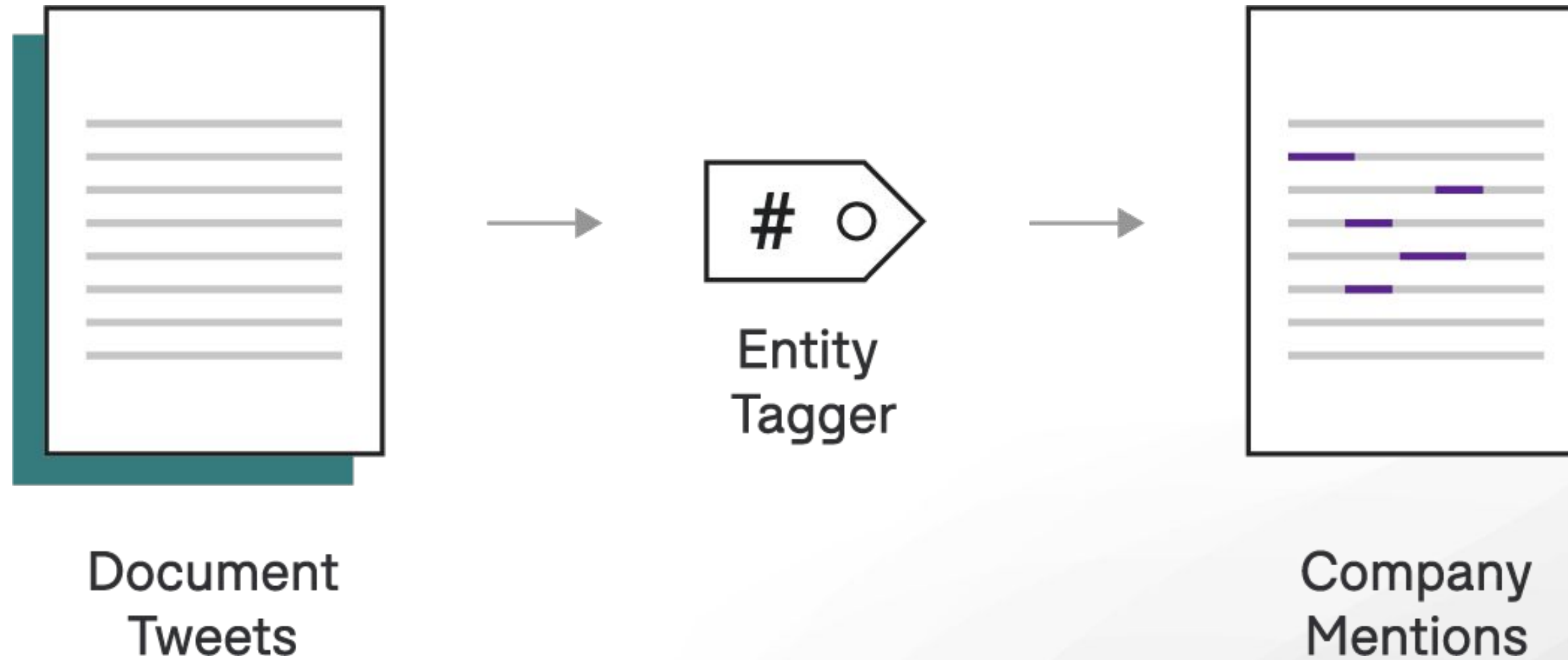
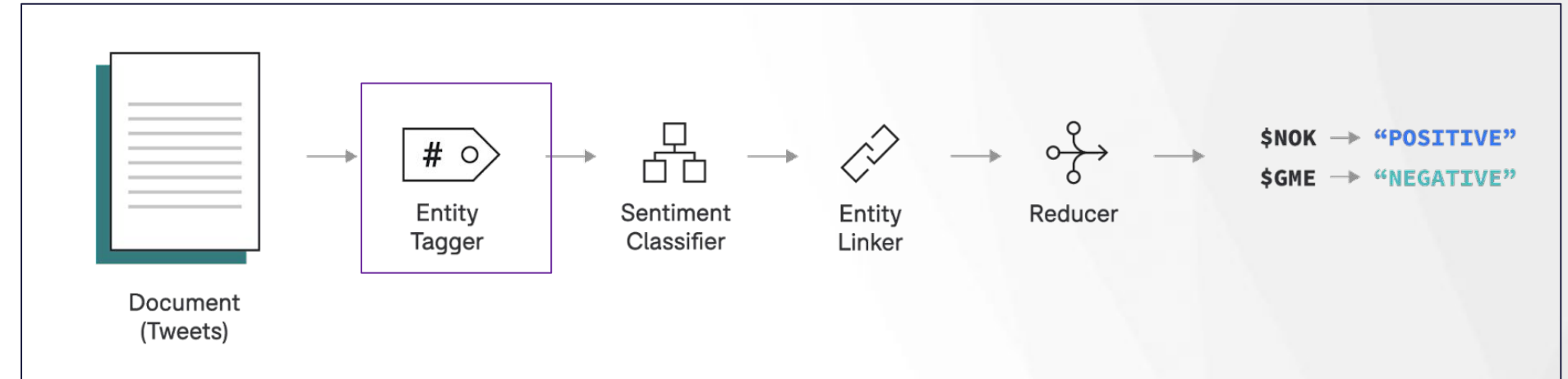
Decompose!



Named Entity Tagger

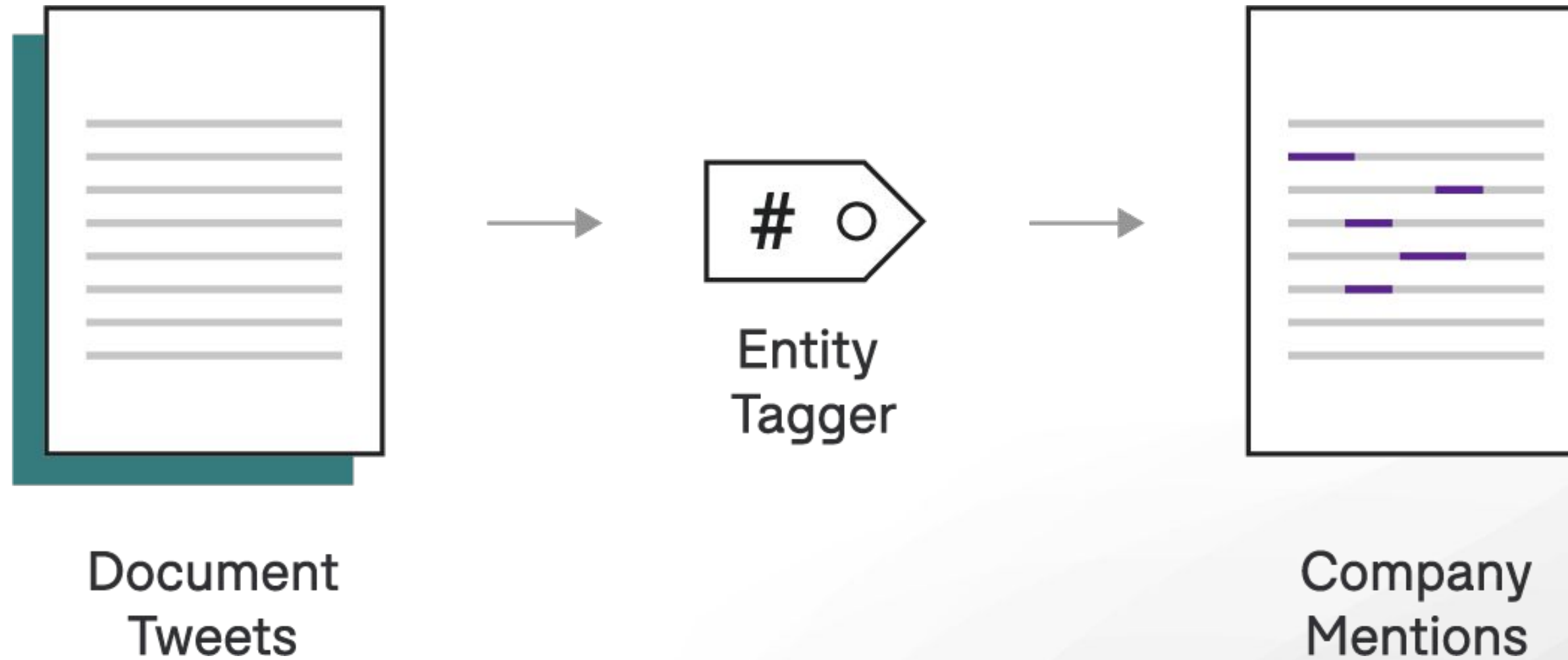
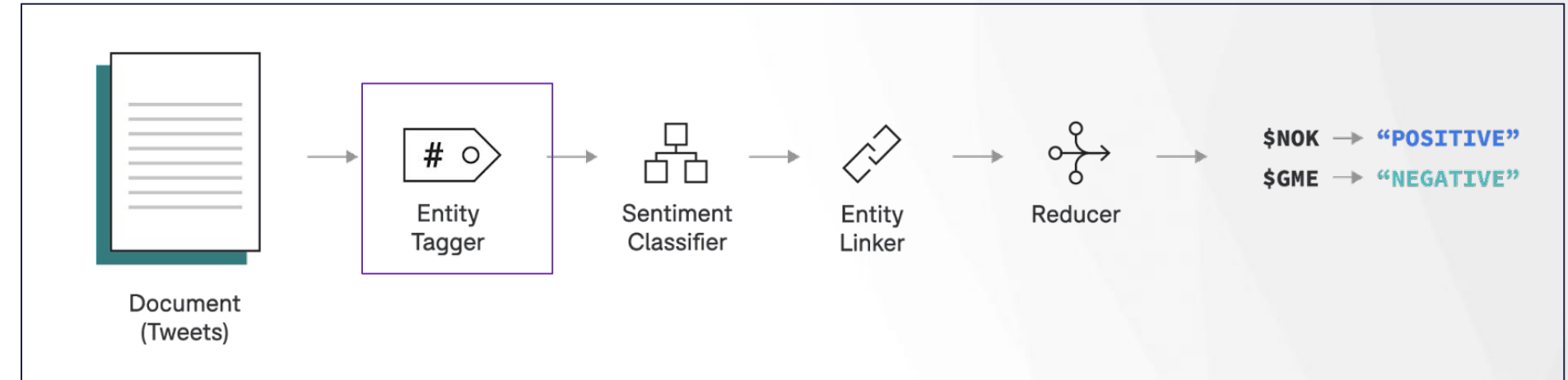


Named Entity Tagger



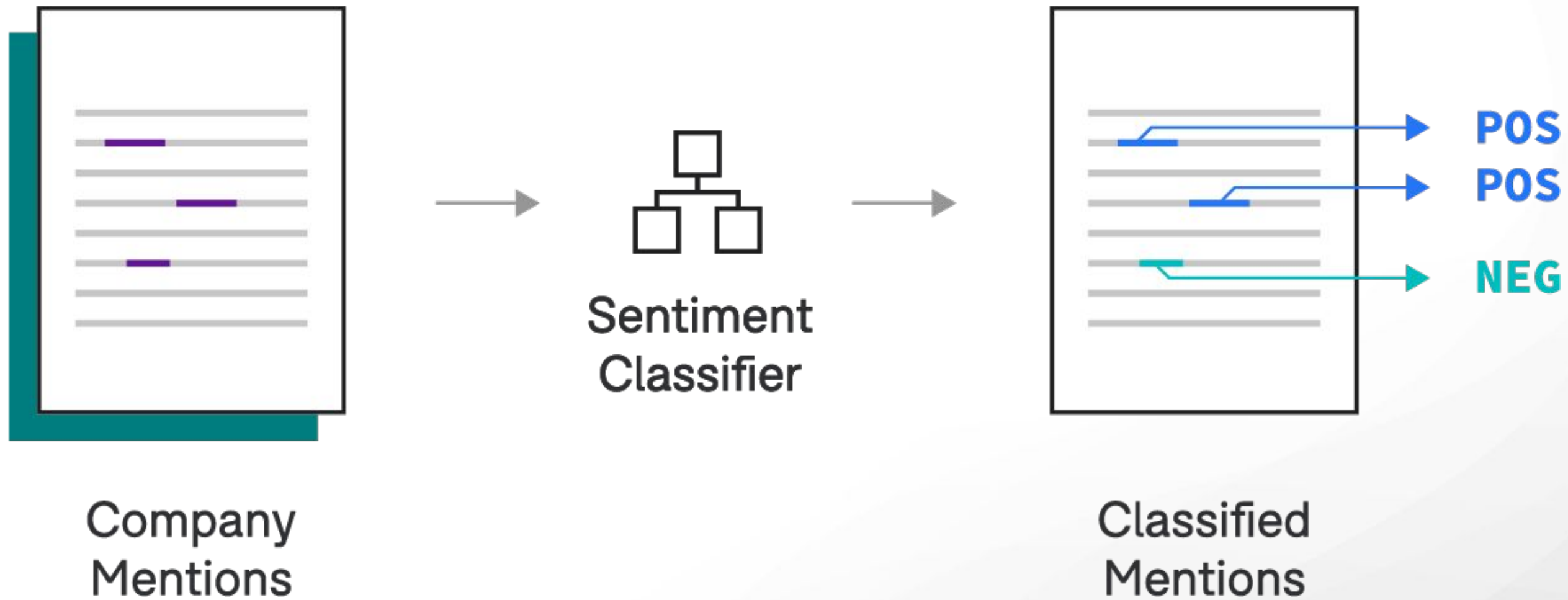
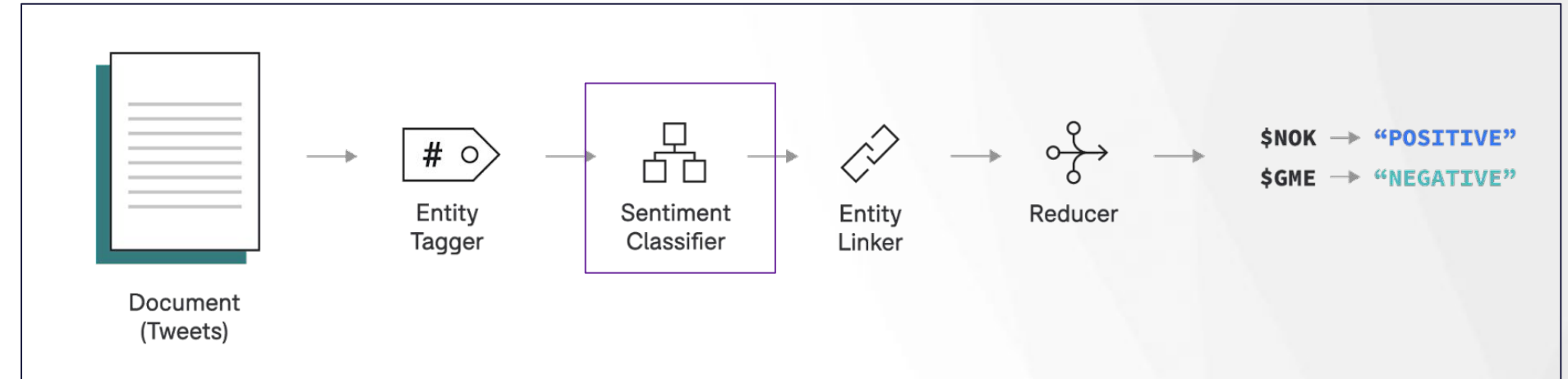
“I scraped a list of Fortune 500 companies last quarter!”

Dictionary-based Named Entity Tagger

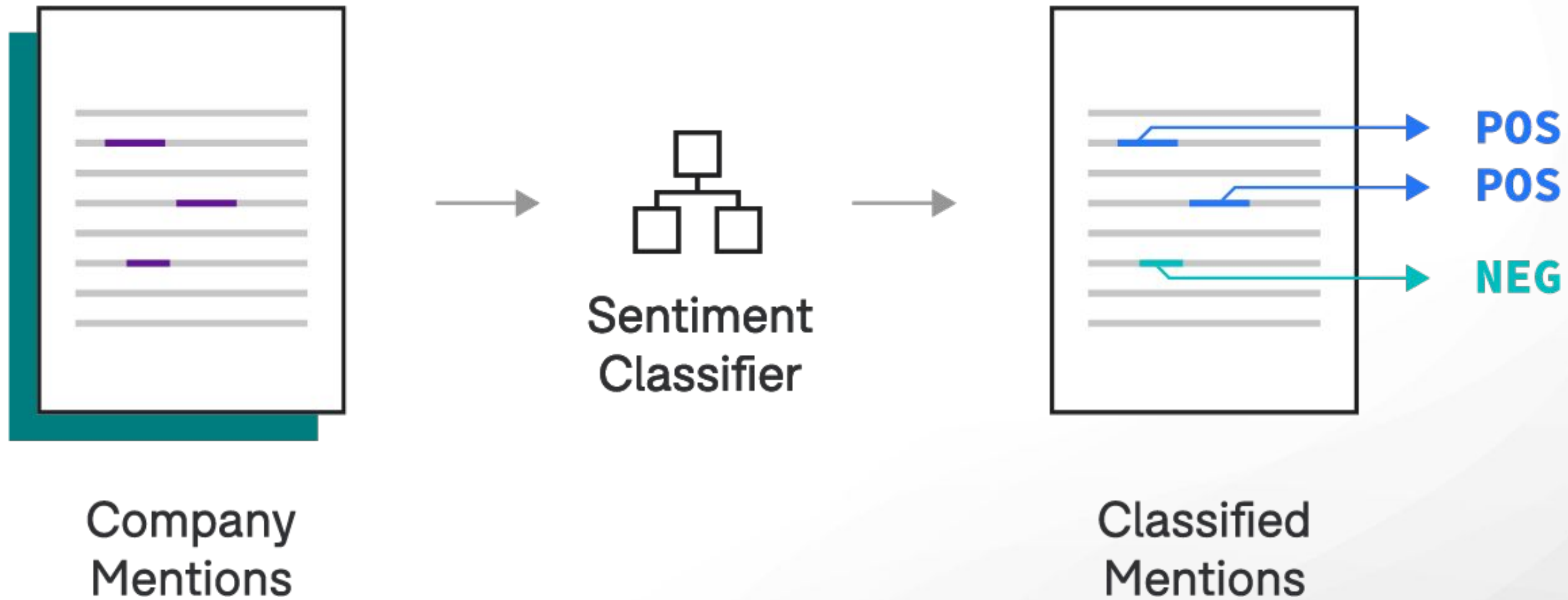
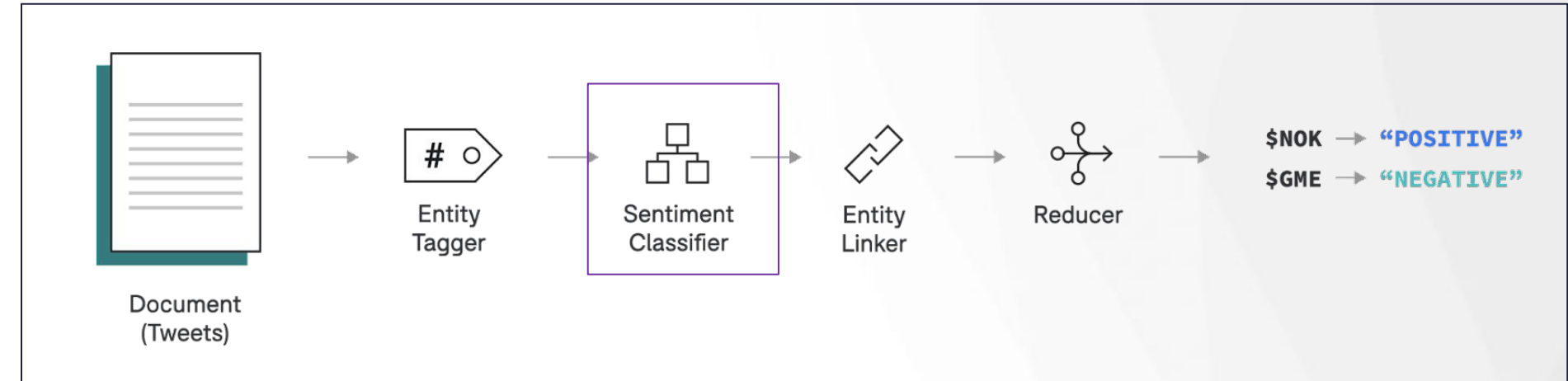


Extract spans based on dictionary keyword matches

Sentiment Classifier

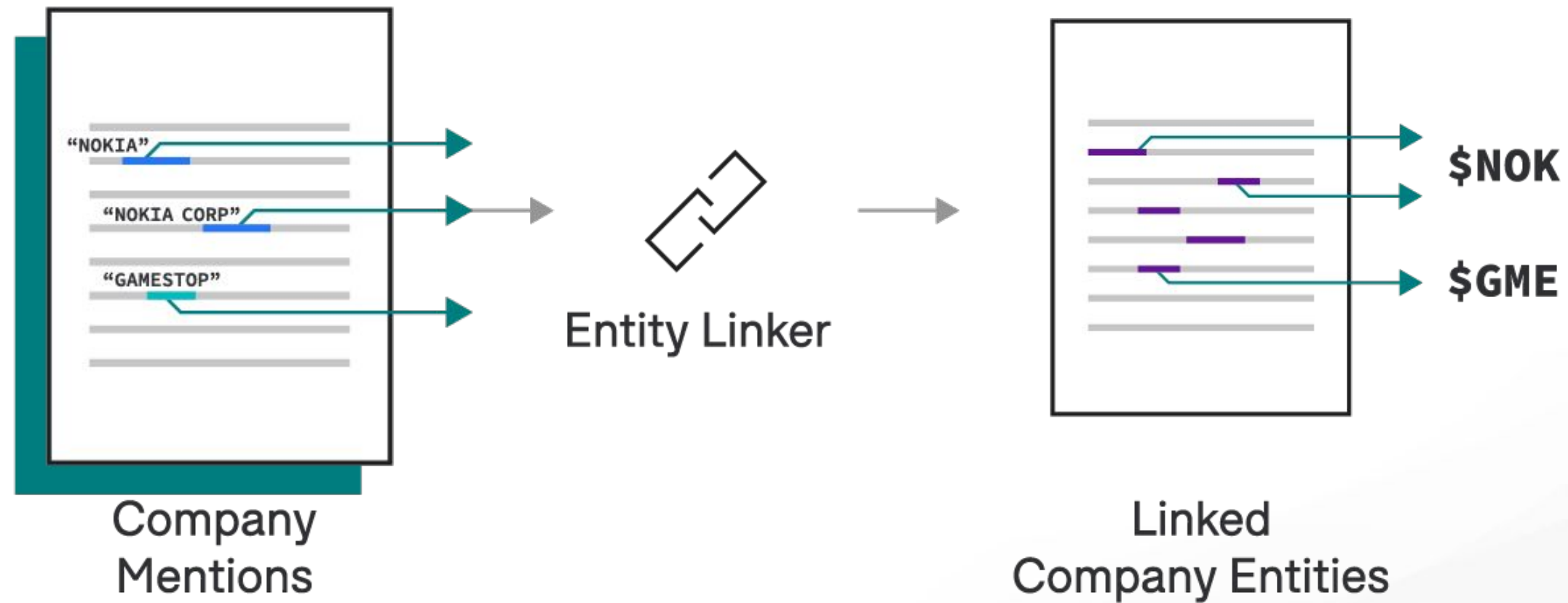
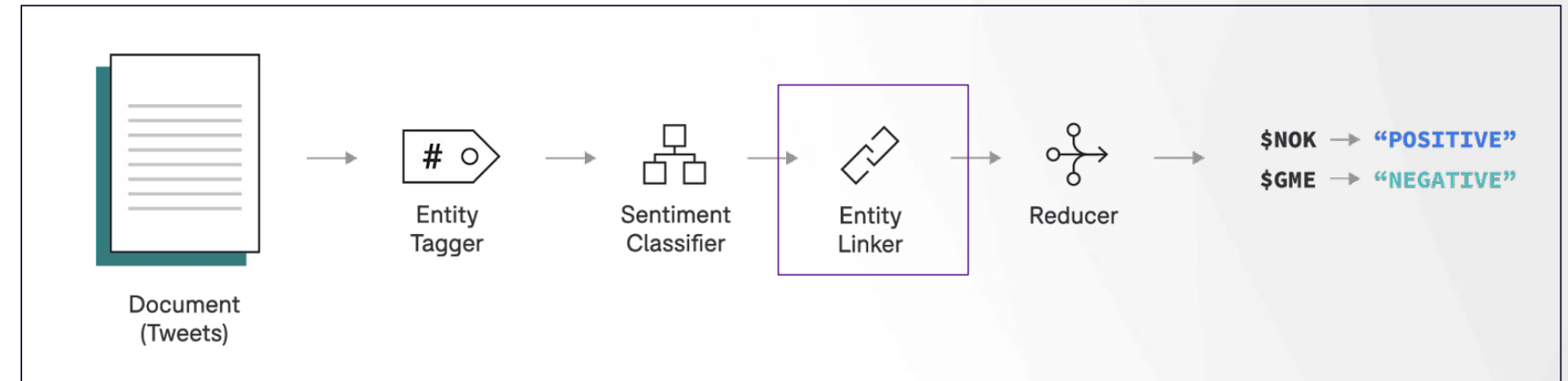


Off-the-shelf Sentiment Classifier

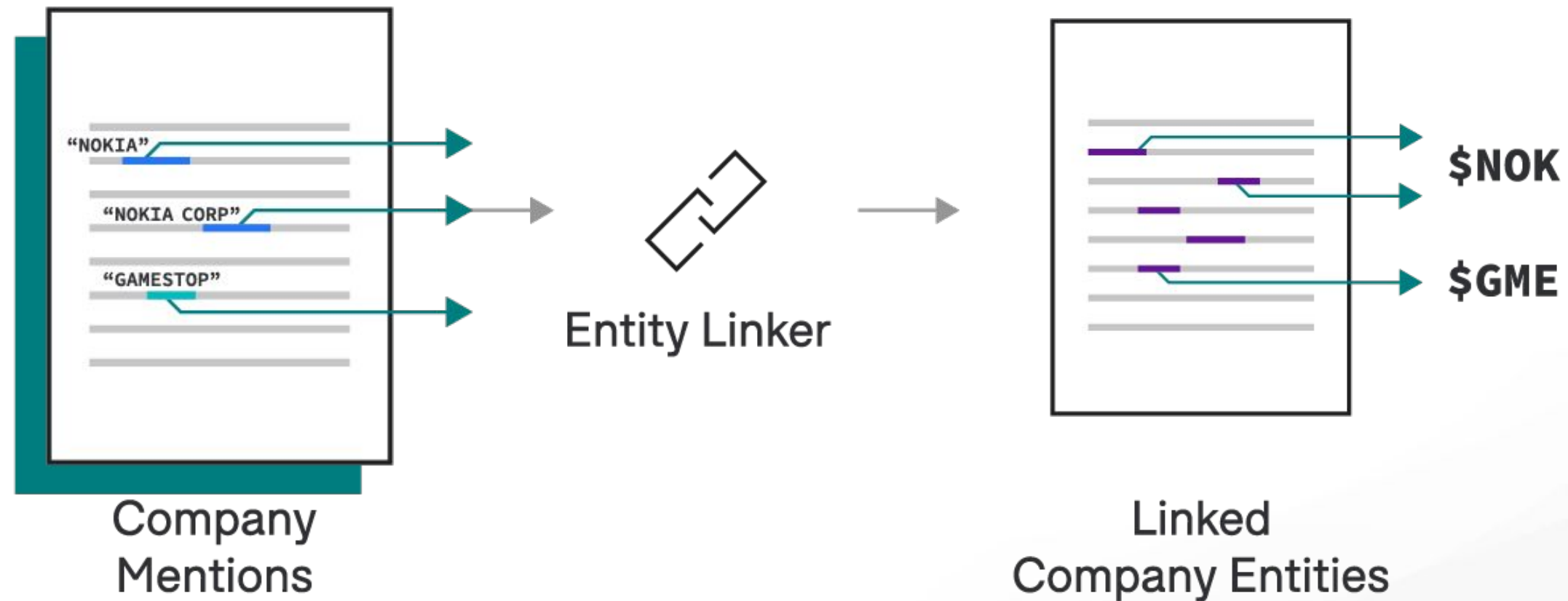
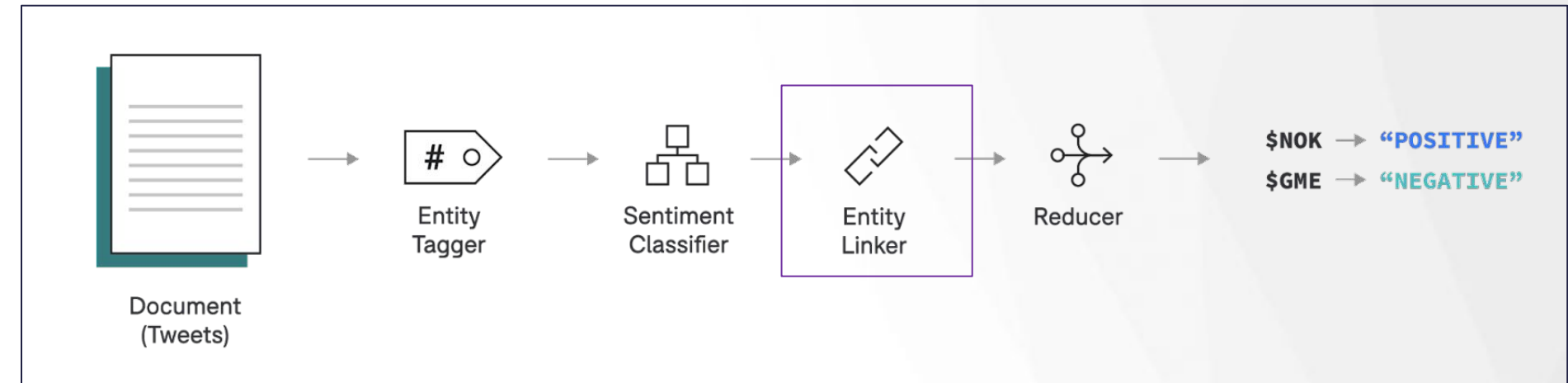


Classify sentiment using contextual words + off-the-shelf model

Entity Linker

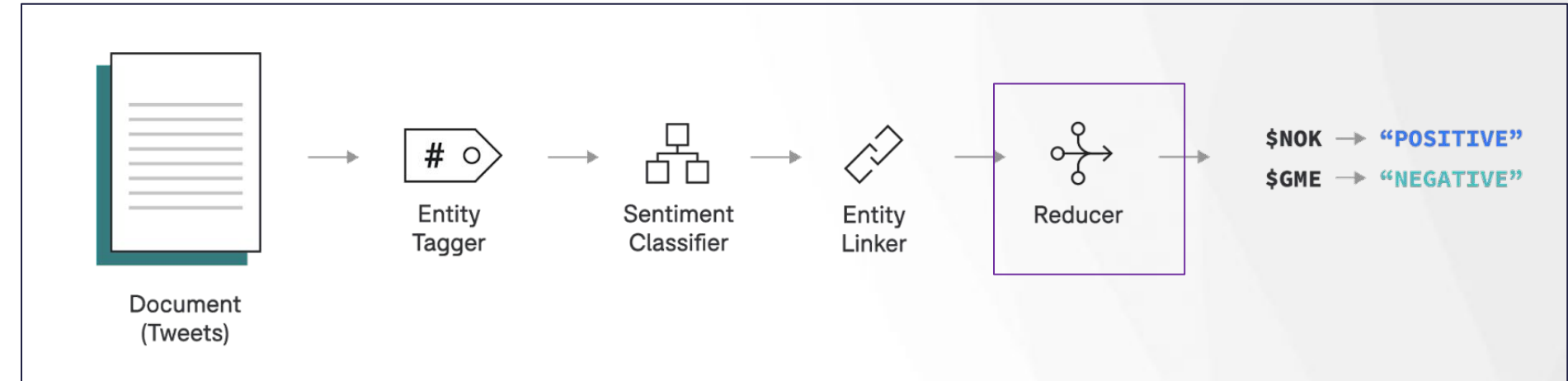


Fuzzy-matching Entity Linker



Fuzzy match to link company mentions to standard stock tickers

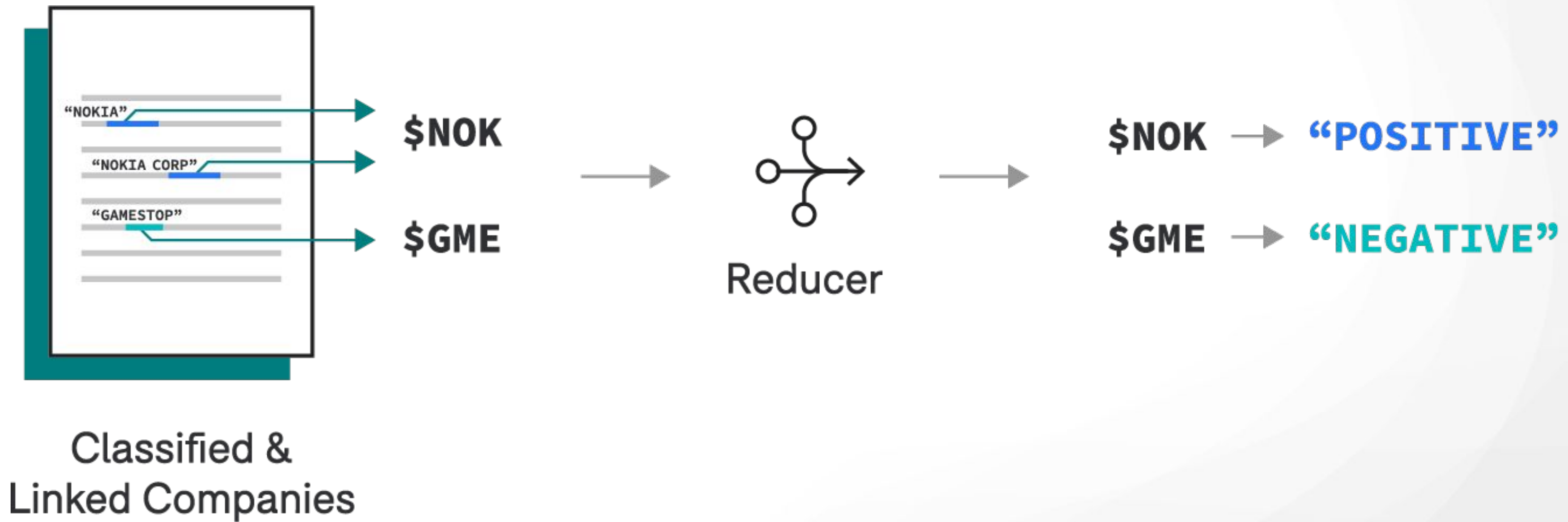
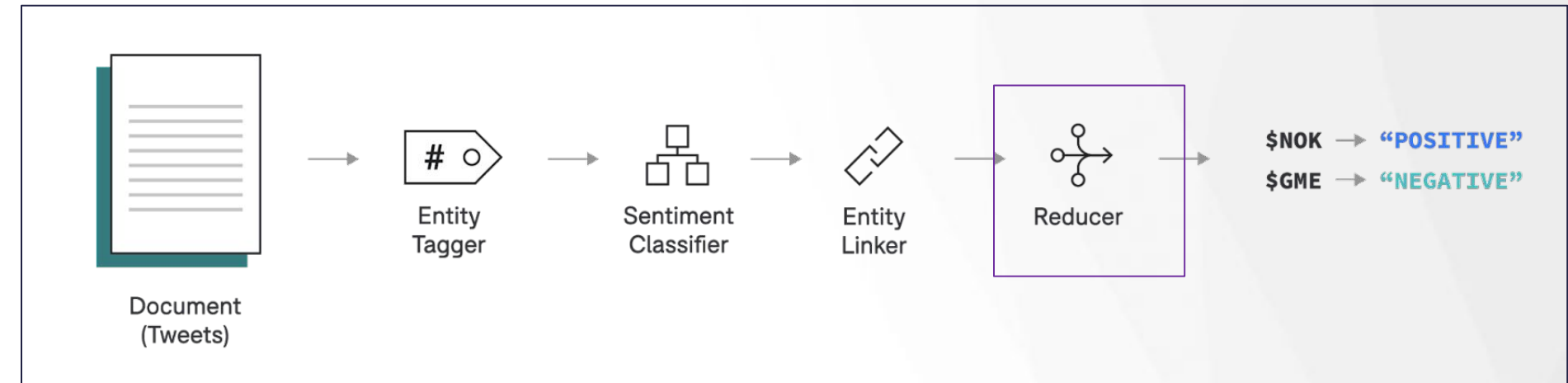
Reducer



Classified & Linked Companies

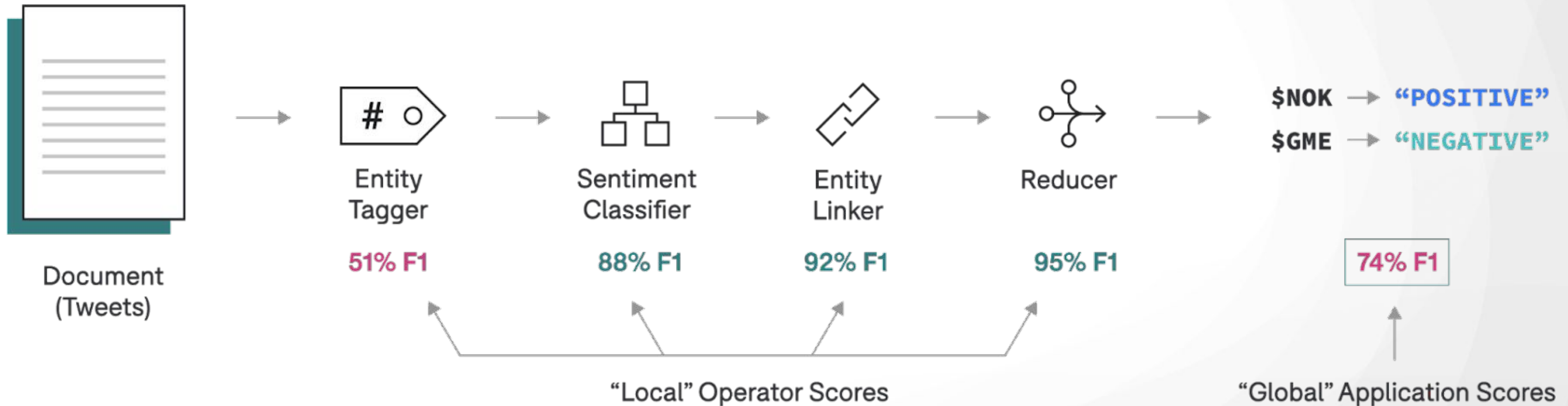


Majority Vote Reducer

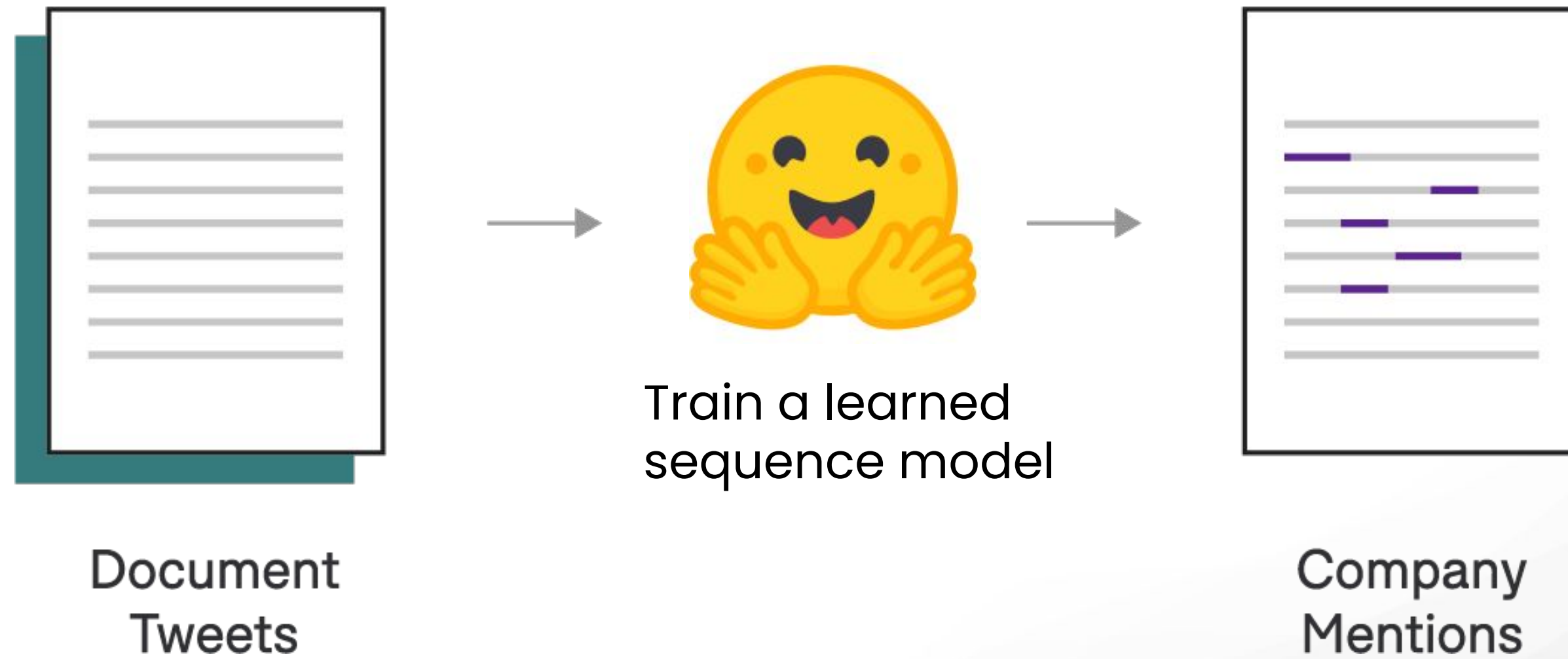


Postprocessor to take the most common sentiment prediction per entity

How do we know where to focus our attention?



Swap out Named Entity Tagger...



Replace dictionary with learned sequence tagging model!

Iterate on sequence tagging data with programmatic labeling

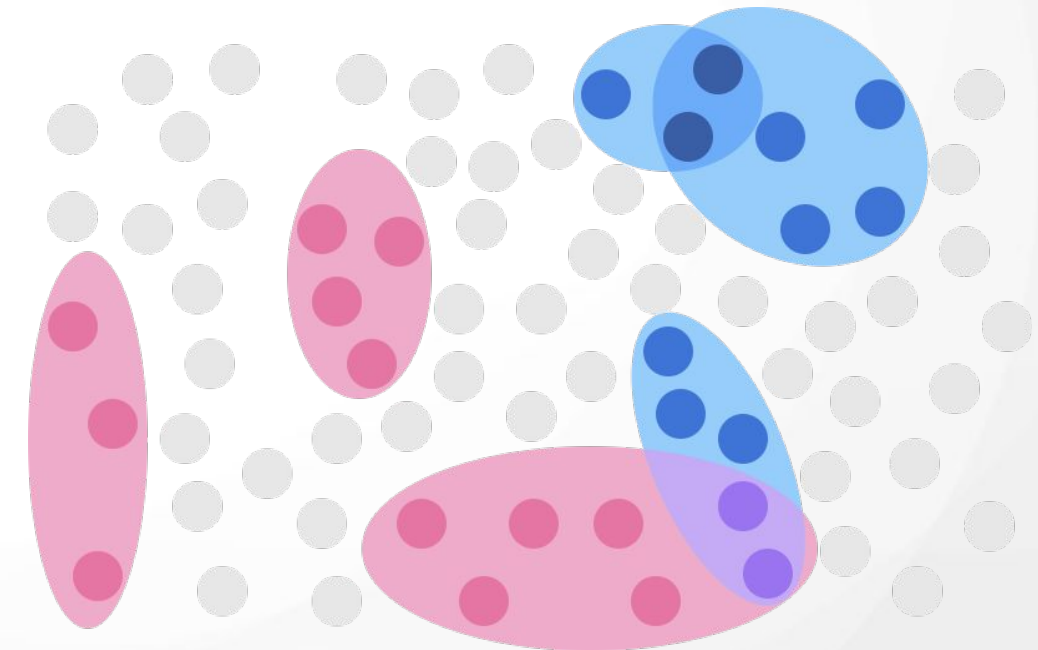
(o*) If span ends with "Inc|Co"...



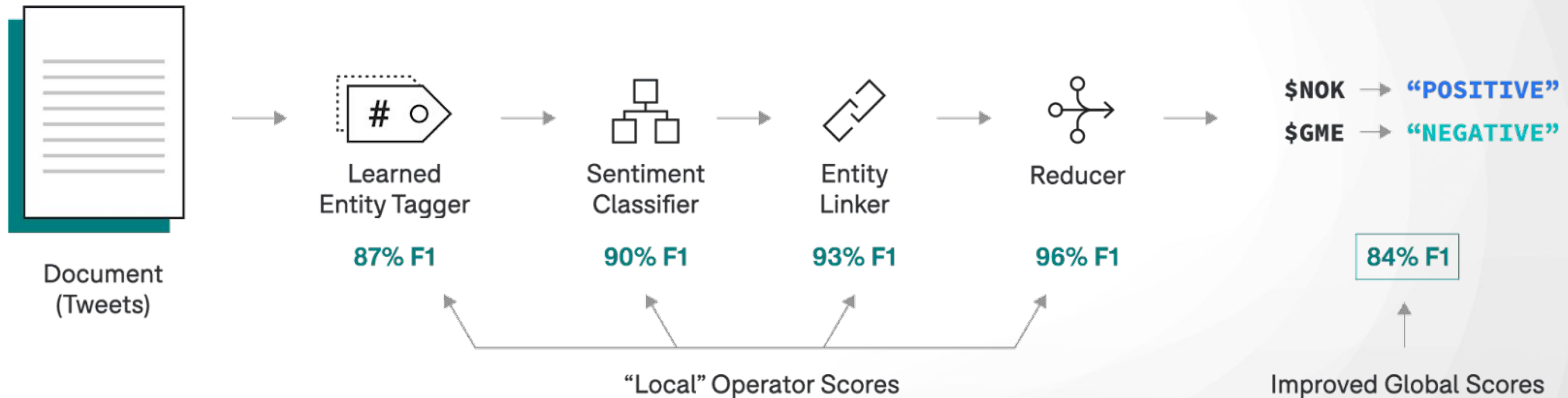
If span matches F500_dict.json...







If span returned by ZSL Model with prompt "What are the named companies?"...



After component-wise debugging, our E2E scores improve!



Data-centric principles for AI engineering

-  **Down with the end-to-end mega model!**
-  **Long live end-to-end (evaluation and iteration)**
-  **ML should not be the universal default**
-  **Rapidly iterate with programmatic labeling**

Data-centric principles for AI engineering software engineering



Down with the end-to-end mega model!

*** single responsibility principle / modularity**



Long live end-to-end (evaluation and iteration)

*** debuggability + introspection**



ML should not be the universal default

*** start simple! avoid premature optimization...**



Rapidly iterate with programmatic labeling

*** anticipate change / incremental development**