

What Do Users Do with LLM-Powered Data Systems?

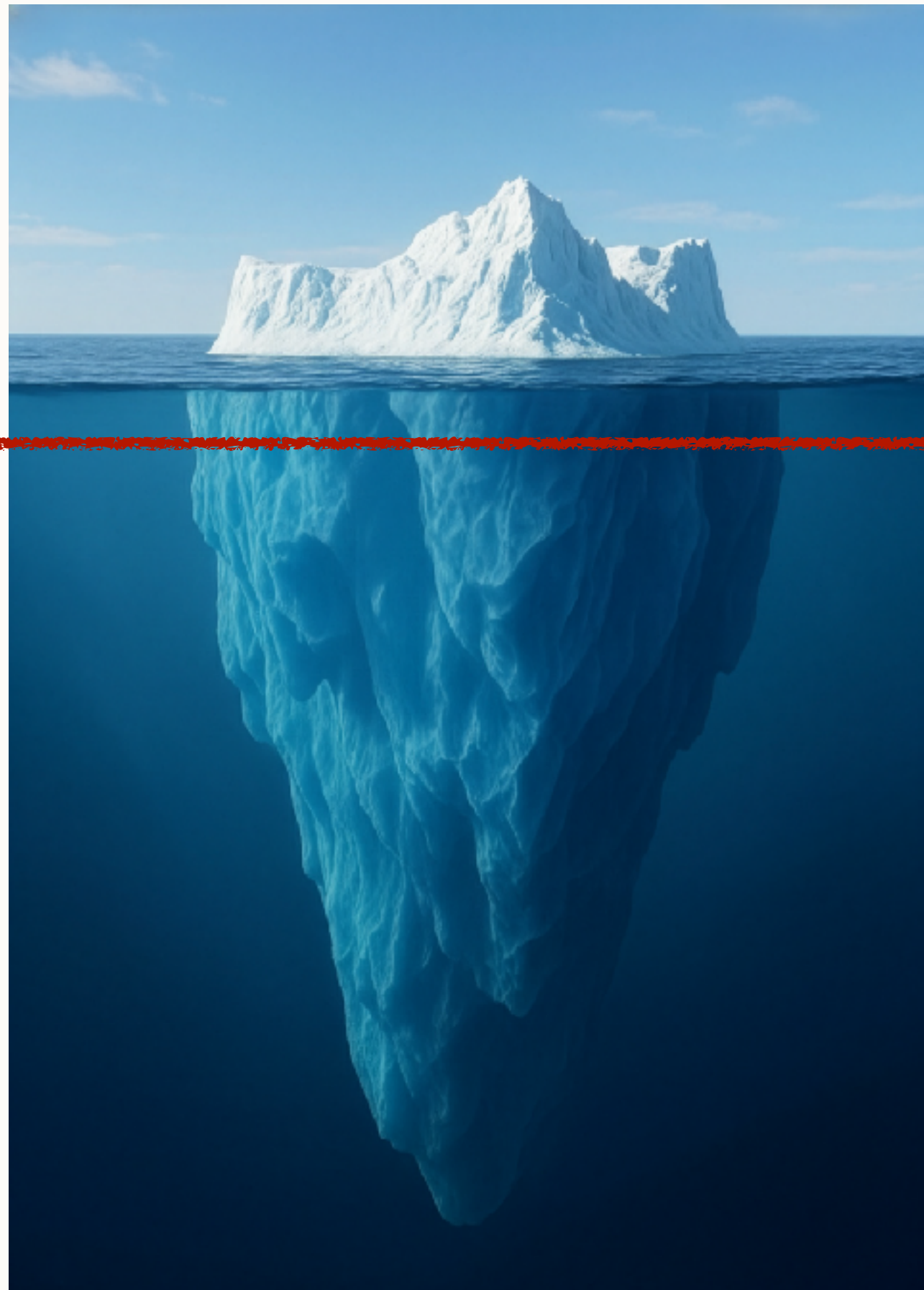
Shreya Shankar

Carnegie Mellon University (formerly UC Berkeley)

September 2026

sh-reya.com

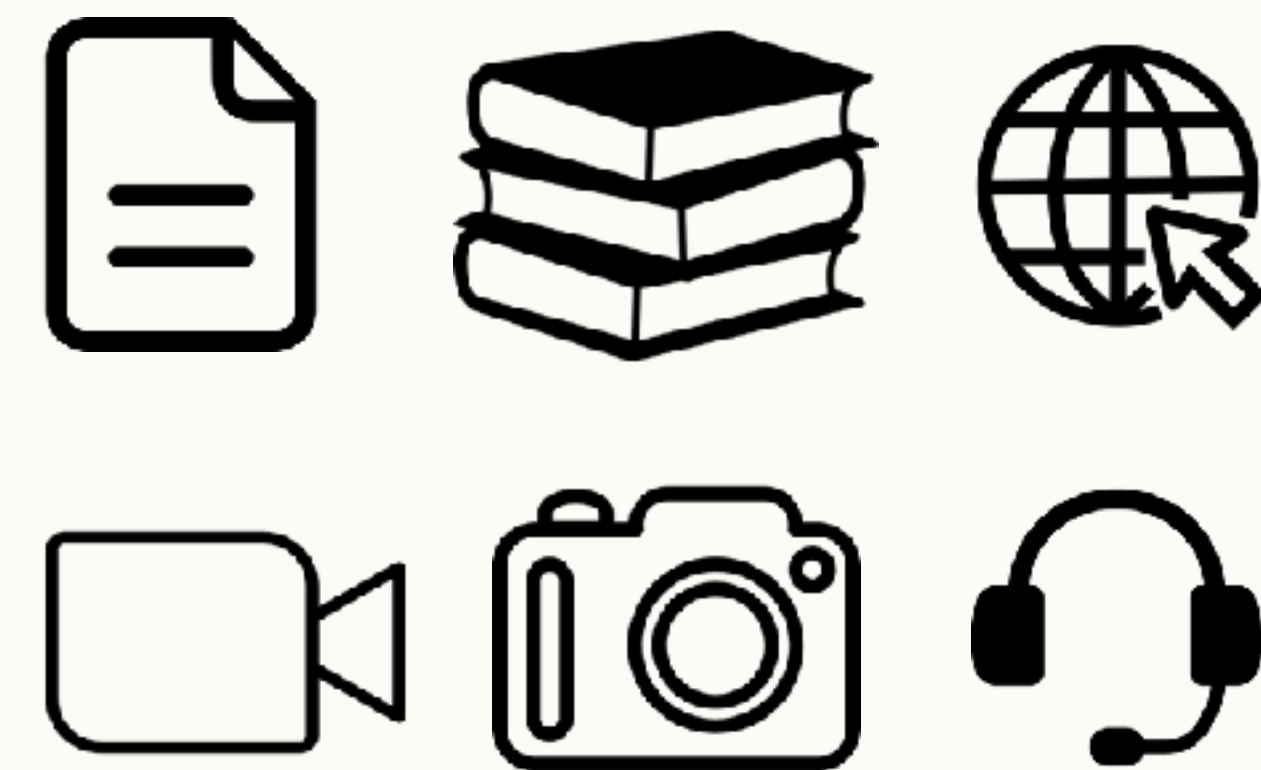
Next-Generation Data Systems



Structured data



Unstructured data



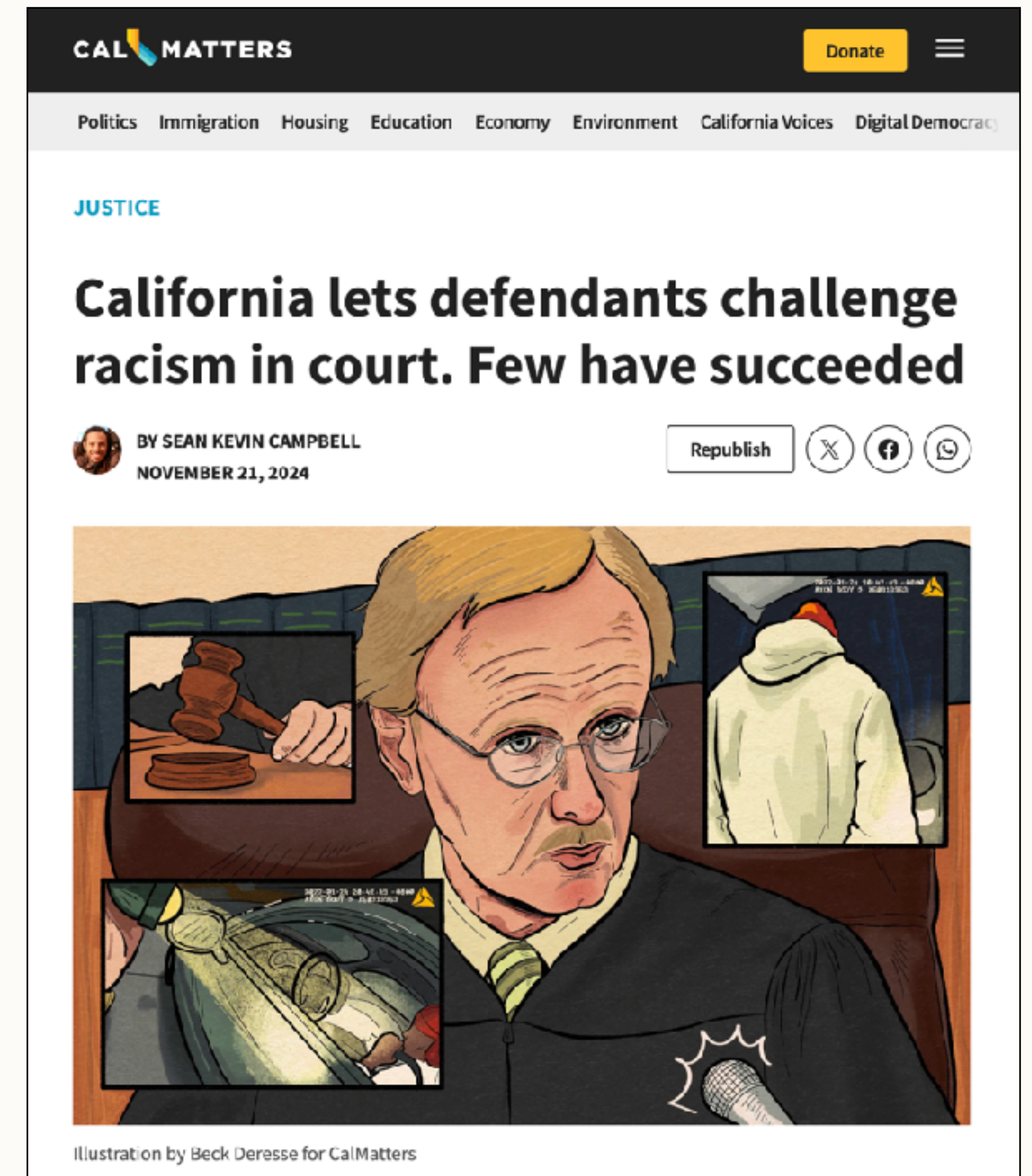
← Can't use SQL to query this type of data

Use Case: Public Defenders

California's Racial Justice Act (2020):
Defendants can challenge convictions for racial bias

Analysis requires: extract evidence from court documents; compare across cases. Millions of documents a year.

Only ~12 successful challenges in 4 years (out of ~10k eligible) because the task is so hard!



Use Cases are Everywhere



What side effects do patients report for Ozempic?

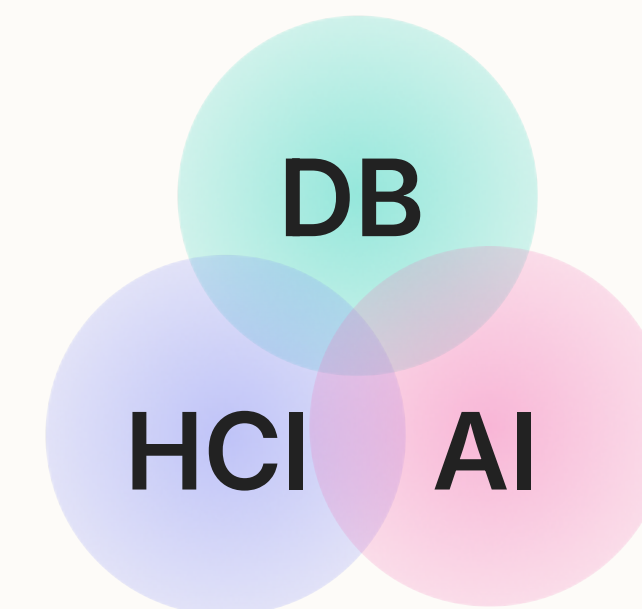
Which contracts have unfavorable arbitration clauses?



What are common complaints in customer reviews?

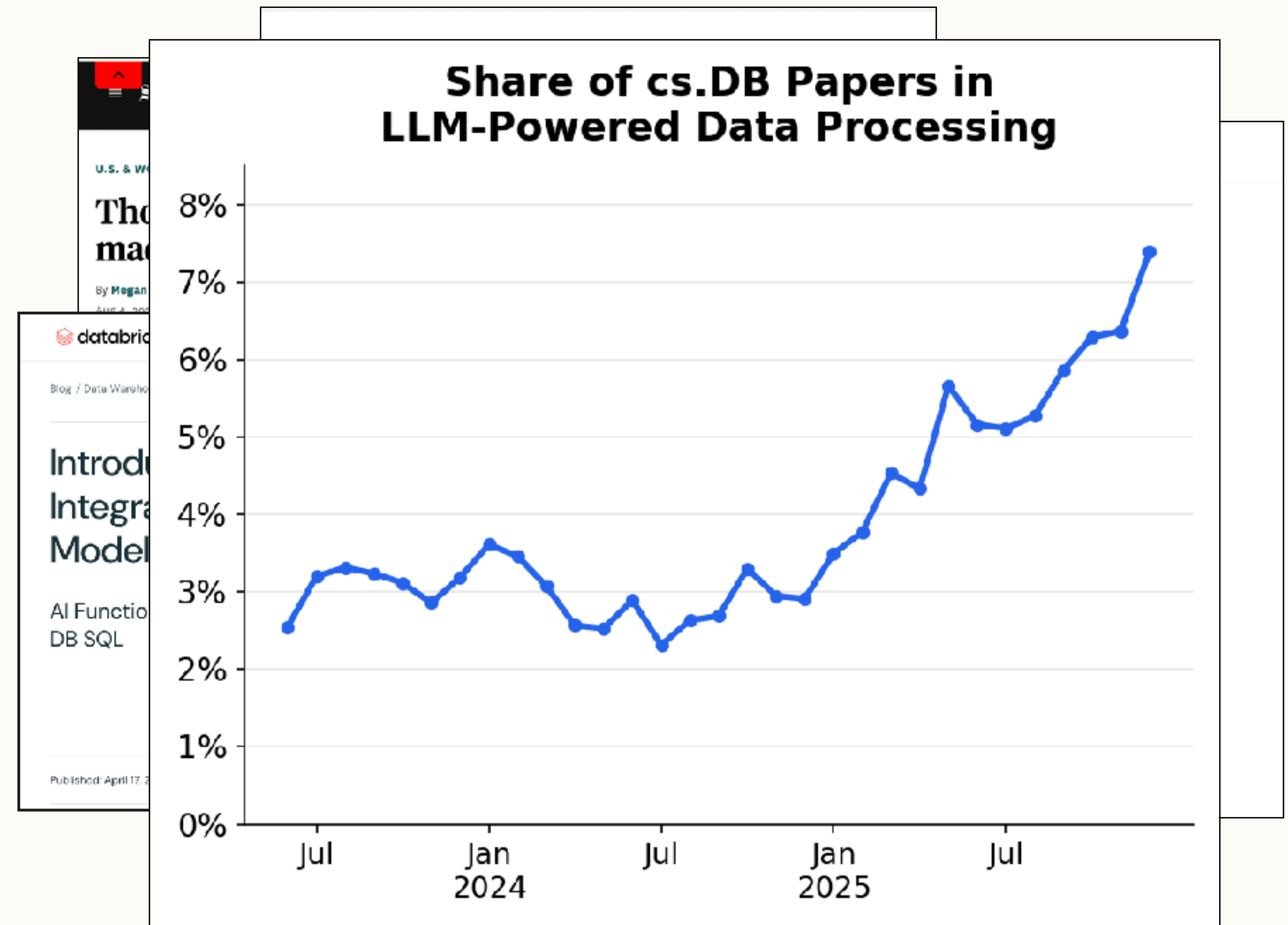
Interdisciplinary challenges:

- ◆ Multi-operation, need to run at scale
- ◆ Ambiguous operation definitions
- ◆ Human-defined; context-dependent



LLM-Powered Data Processing is Taking Off

- ◆ Real and high-stakes deployments: e.g., journalists, public defenders, climate scientists
- ◆ Industry adoption: Snowflake, BigQuery, Databricks
- ◆ Growing research community
- ◆ DocETL: a system we built at UC Berkeley.



Today

1. Example DocETL query
2. Why authoring DocETL (and LLM-powered) queries is hard for users
3. DocWrangler, our IDE for DocETL
 - i. What we built, and why
 - ii. What we learned

Example: DocETL Pipeline/Query

type: `map`

prompt: “Determine the name of the judge in this trial”

output:

schema:

`name`: `str`



type: `reduce`

reduce_key: `name`

prompt: “Summarize the common implicit biases shown by this judge across all their trial transcripts”

output:

schema:

`common_biases`: `str`

id	transcript	<code>name</code>
...	...	...

<code>name</code>	<code>common_biases</code>
...	...

HCI Problems with the DocETL Paradigm

Users struggle to write good queries!

type: `map`

prompt: “Determine the name of the judge in this trial”

output:

schema:

`name`: str



type: `reduce`

reduce_key: name

prompt: “Summarize the common implicit biases shown by this judge across all their trial transcripts”

output:

schema:

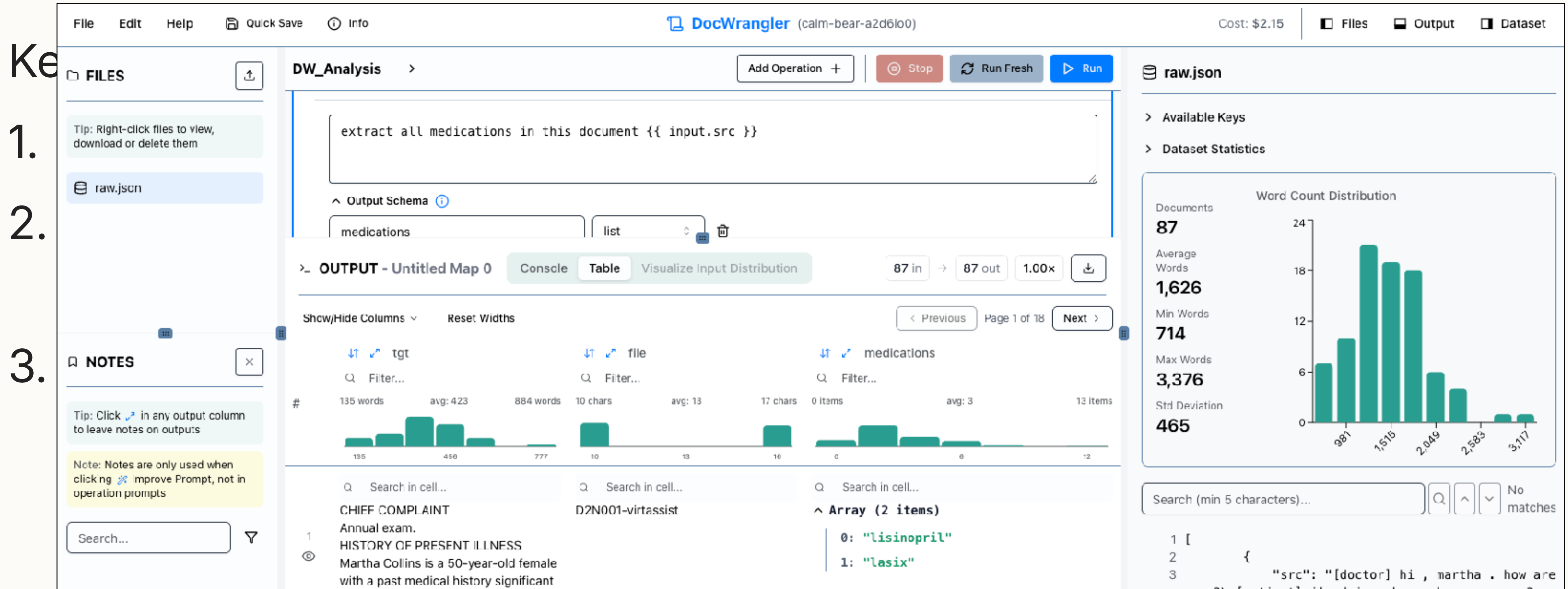
`common_biases`: str

Research question: How can users *steer* LLM-powered unstructured data analysis?

DocWrangler

Steering Semantic Data Processing with DocWrangler. **Shankar***, Chopra* et al. *UIST '25*. 🏆

A mixed-initiative IDE for DocETL.



FILES

Tip: Right-click files to view, download or delete them

raw.json

NOTES

Tip: Click  in any output column to leave notes on outputs

Note: Notes are only used when clicking  Improve Prompt, not in operation prompts

Search...

Untitled_Analysis

<

OverviewSystem Prompts15

Add Operation +

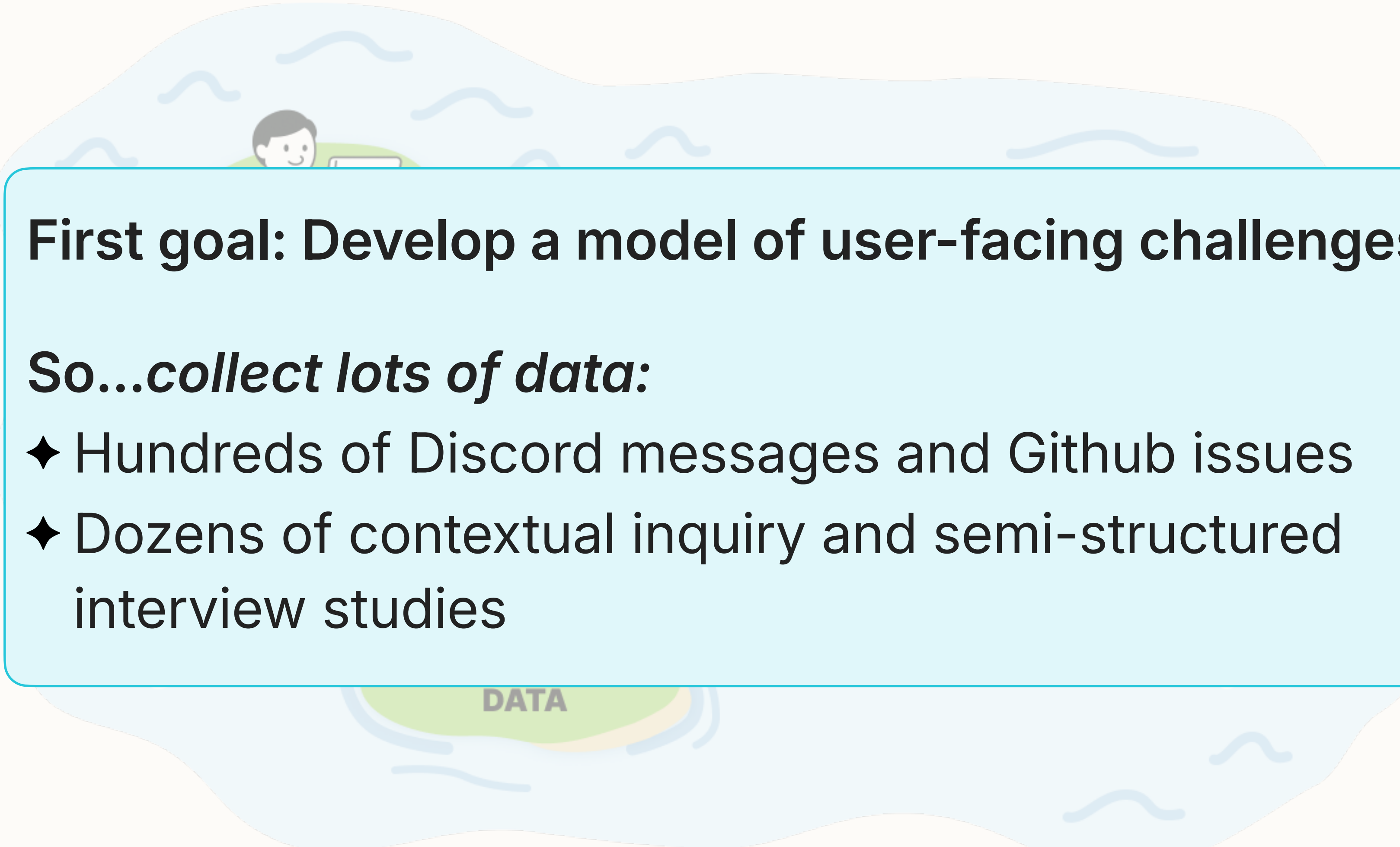
StopRun FreshRun

+ Add Operation

Speed > 20x

Key Idea: Three Gulfs Framework

Steering Semantic Data Processing with DocWrangler. **Shankar***, Chopra* et al. *UIST '25*. 🏆



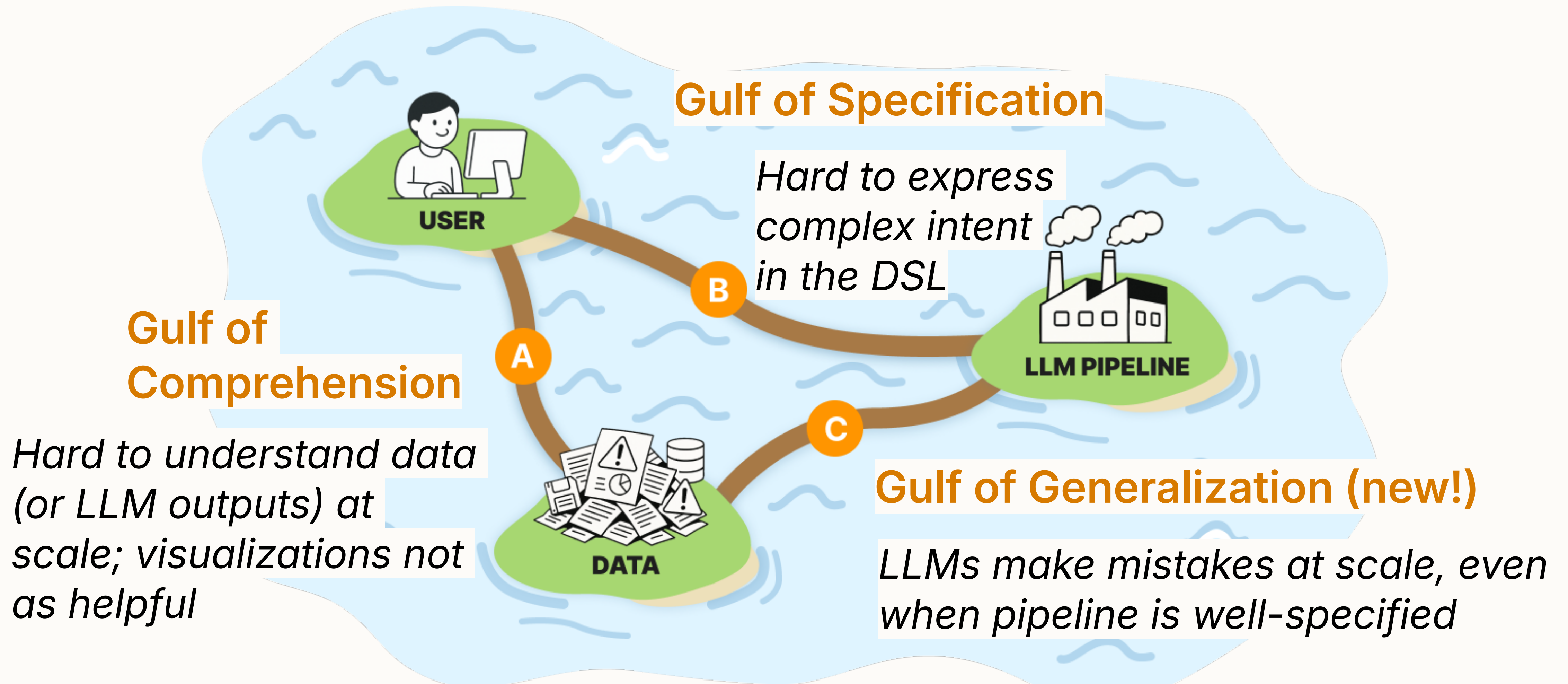
First goal: Develop a model of user-facing challenges.

So...collect lots of data:

- ◆ Hundreds of Discord messages and Github issues
- ◆ Dozens of contextual inquiry and semi-structured interview studies

Key Idea: Three Gulfs Framework

Steering Semantic Data Processing with DocWrangler. **Shankar***, Chopra* et al. *UIST '25*. 🏆



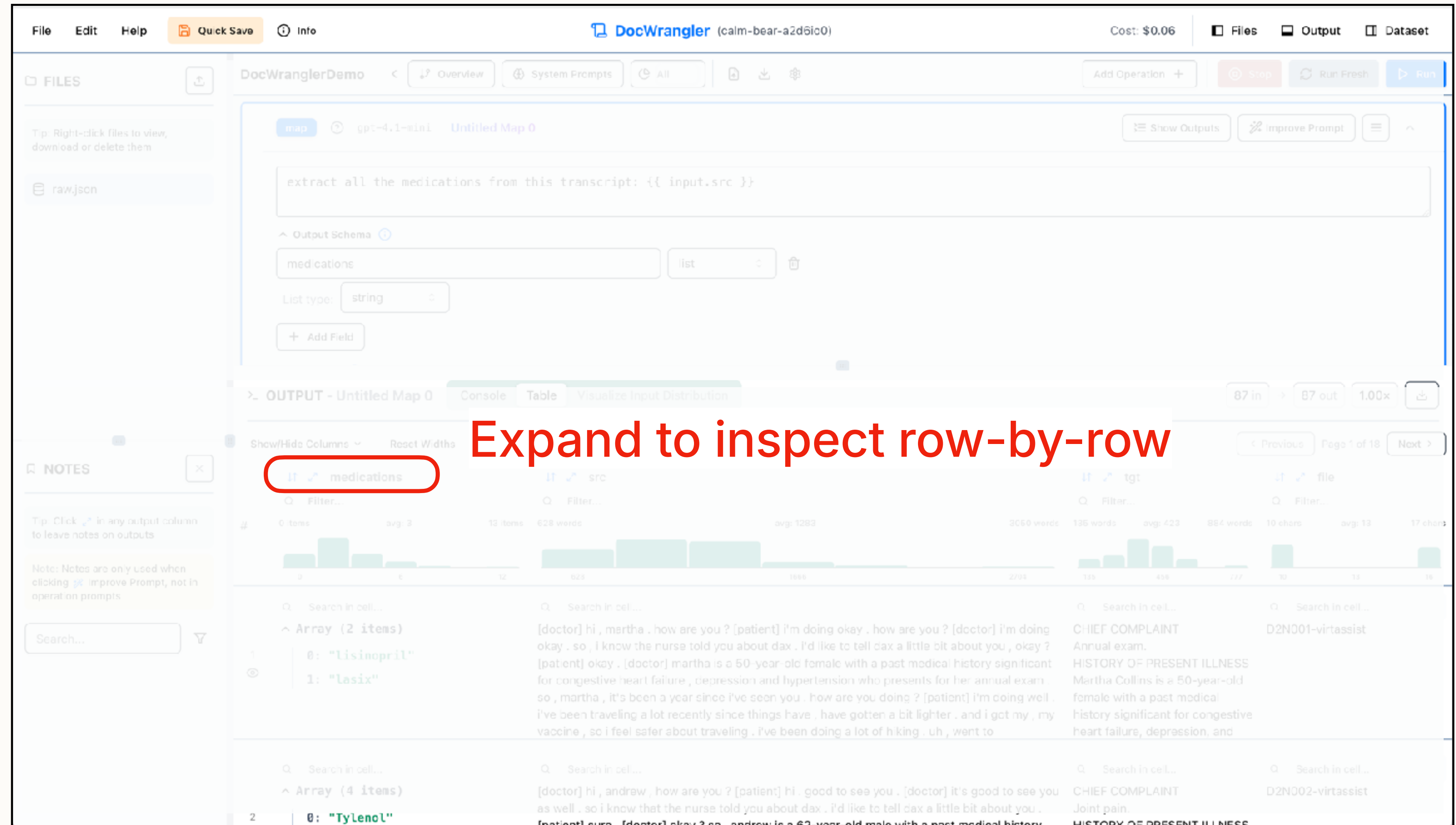
Illustrating the Gulf of Comprehension

Task

"Extract all  from each transcript"

Challenge

Discovering what you really want & the long tail of LLM behaviors



The screenshot shows the DocWrangler interface. At the top, there's a menu bar with File, Edit, Help, Quick Save, and Info. The main header displays 'DocWrangler (calm-bear-a2d6ic0)' and 'Cost: \$0.06'. Below this, there's a 'FILES' section on the left with a tip: 'Tip: Right-click files to view, download or delete them'. The main area shows a 'map' operation with the prompt 'extract all the medications from this transcript: {{ input.src }}'. The 'Output Schema' section shows a field named 'medications' with a 'list' type and a 'string' list type. Below this, there's a 'Table' view of the output. The table has columns for 'medications', 'src', 'tgt', and 'file'. The 'medications' column is highlighted with a red box and the text 'Expand to inspect row-by-row'. The table shows two rows of data. The first row has a 'medications' value of an array containing 'Lisinopril' and 'lasix'. The second row has a 'medications' value of an array containing 'Tylenol'.

Expand to inspect row-by-row

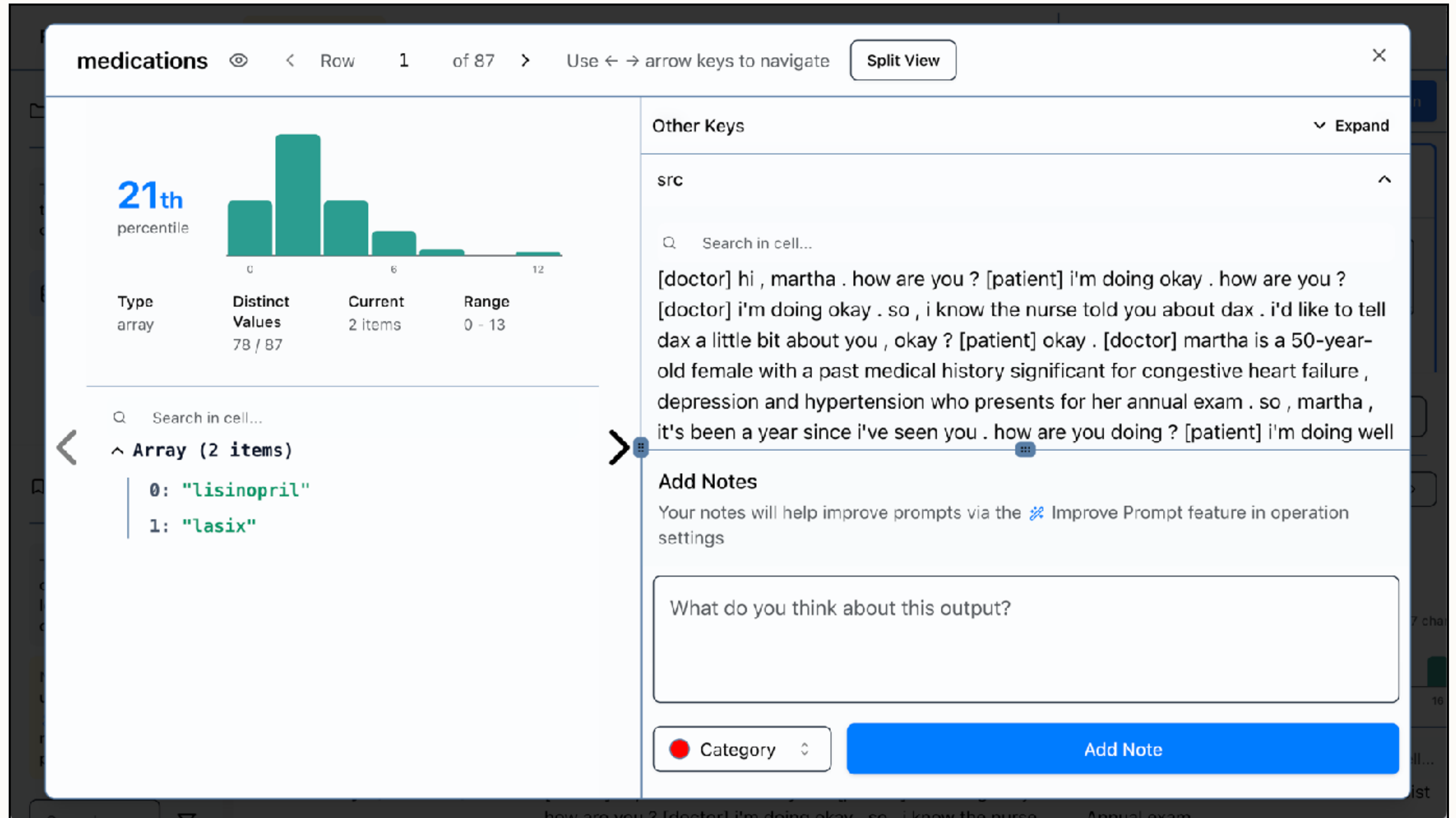
In-Situ Notetaking to *Comprehend* Data Better

Task

"Extract all  from each transcript"

Challenge

Discovering what you really want & the long tail of LLM behaviors



The screenshot displays a data analysis interface for a dataset named "medications". At the top, it shows "Row 1 of 87" and navigation instructions. A "Split View" button is present. The main area is divided into two panels. The left panel features a histogram with a peak at value 6, labeled "21th percentile". Below the histogram, summary statistics are listed: Type: array, Distinct Values: 78 / 87, Current: 2 items, and Range: 0 - 13. Below these, a search bar and an expanded view of the array are shown, containing two items: "lisinopril" and "lasix". The right panel displays a chat transcript under the heading "Other Keys" with an "Expand" button. The transcript shows a conversation between a doctor and a patient. Below the transcript, there is an "Add Notes" section with a text input field containing the prompt "What do you think about this output?". At the bottom of the right panel, there is a "Category" dropdown menu and a blue "Add Note" button.

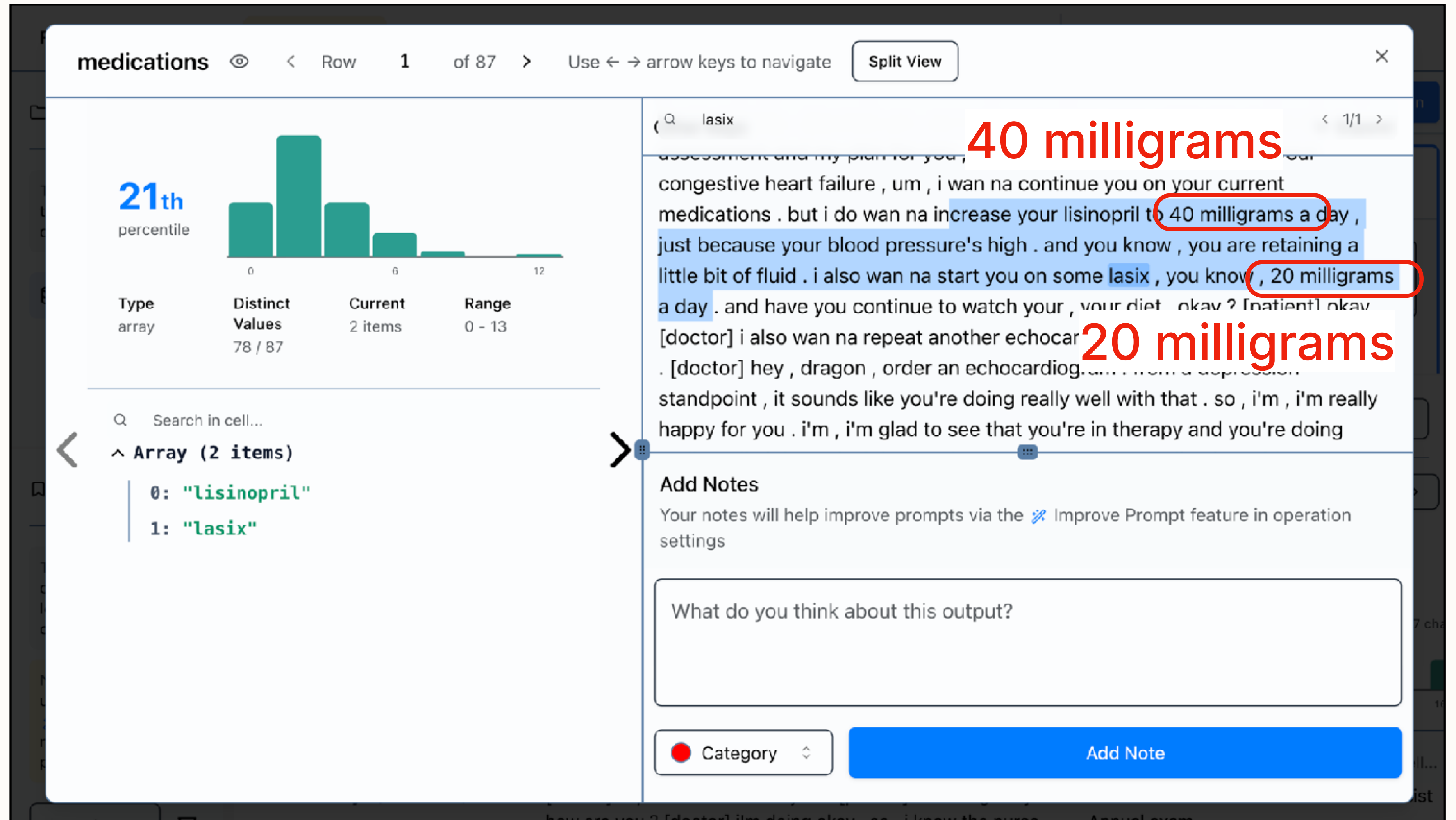
In-Situ Notetaking to *Comprehend* Data Better

Task

"Extract all  from each transcript"

Challenge

Discovering what you really want & the long tail of LLM behaviors



The screenshot displays a data visualization tool interface. On the left, a histogram shows the distribution of medication counts, with a peak at 1. Below the histogram, a table lists statistics: Type (array), Distinct Values (78 / 87), Current (2 items), and Range (0 - 13). A search bar labeled 'Search in cell...' is present. Below the search bar, an 'Array (2 items)' is shown with elements: 0: "lisinopril" and 1: "lasix". On the right, a text transcript is displayed, with several phrases highlighted in blue and red. The highlighted phrases are: "40 milligrams", "20 milligrams", and "20 milligrams". Below the transcript, there is a section for 'Add Notes' with a text input field and a blue 'Add Note' button.

Type	Distinct Values	Current	Range
array	78 / 87	2 items	0 - 13

Search in cell...

Array (2 items)

- 0: "lisinopril"
- 1: "lasix"

40 milligrams

20 milligrams

20 milligrams

Add Notes

Your notes will help improve prompts via the [Improve Prompt](#) feature in operation settings

What do you think about this output?

Category 

Add Note

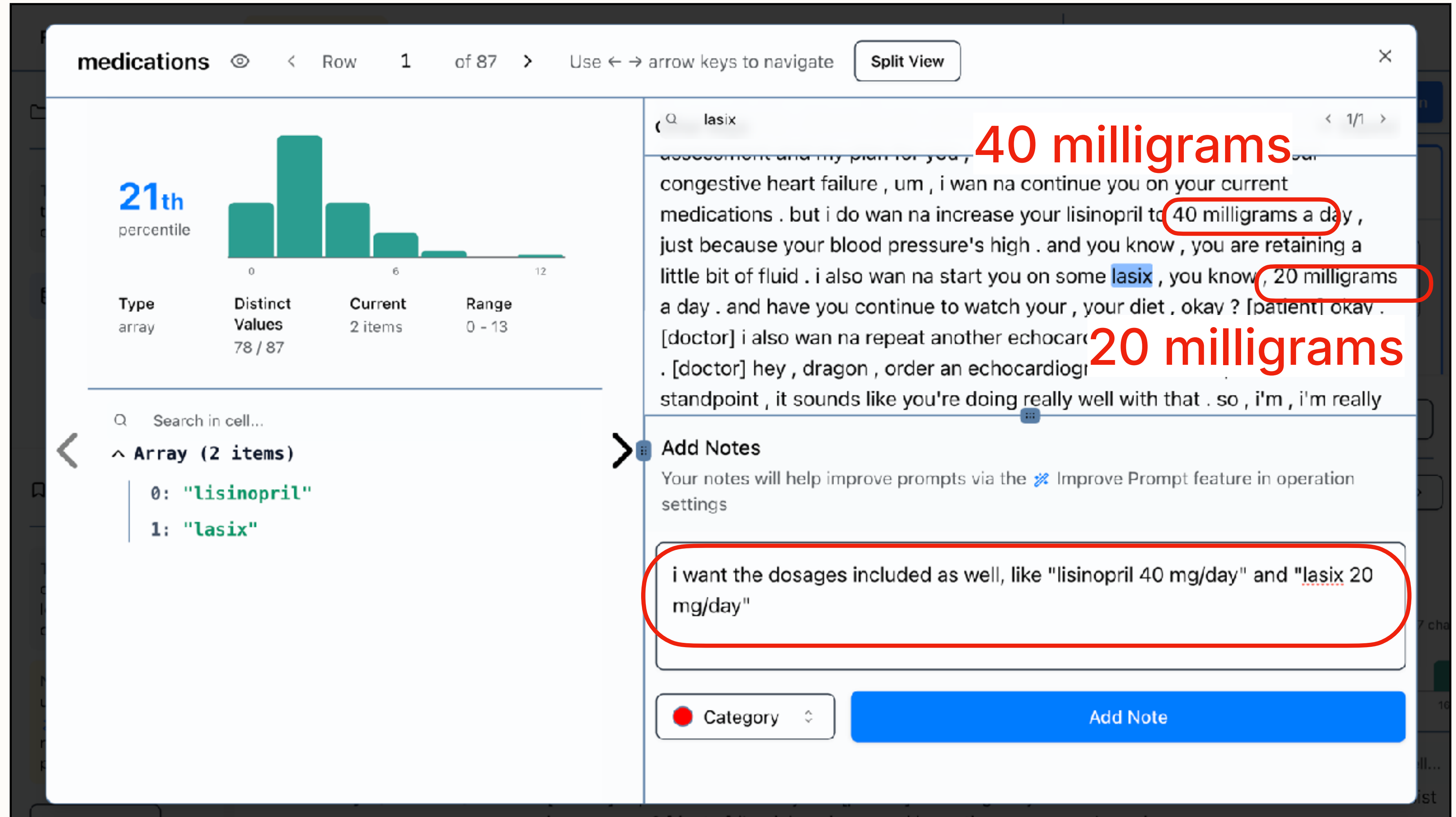
In-Situ Notetaking to *Comprehend* Data Better

Task

"Extract all 
from each
transcript"

Challenge

Discovering
what you
really want &
the long tail of
LLM behaviors



The screenshot displays a data visualization interface for a dataset named "medications". At the top, it shows "Row 1 of 87" and a "Split View" button. The main area is divided into two panels. The left panel features a histogram with a peak at 0, labeled "21th percentile". Below the histogram, a table lists statistics: Type (array), Distinct Values (78 / 87), Current (2 items), and Range (0 - 13). Below this, a search bar "Search in cell..." is followed by an expanded "Array (2 items)" showing "0: 'lisinopril'" and "1: 'lasix'". The right panel shows a transcript of a medical consultation. Handwritten red notes are present: "40 milligrams" above "40 milligrams a day", "20 milligrams" above "20 milligrams", and a larger note "i want the dosages included as well, like 'lisinopril 40 mg/day' and 'lasix 20 mg/day'" in a red box. At the bottom right, there is an "Add Notes" section with a text input field, a "Category" dropdown, and an "Add Note" button.

medications  < Row 1 of 87 > Use ← → arrow keys to navigate Split View

21th percentile

Type array Distinct Values 78 / 87 Current 2 items Range 0 - 13

Search in cell...

Array (2 items)

- 0: "lisinopril"
- 1: "lasix"

lasix

40 milligrams

20 milligrams

i want the dosages included as well, like "lisinopril 40 mg/day" and "lasix 20 mg/day"

Add Notes

Your notes will help improve prompts via the [Improve Prompt](#) feature in operation settings

Category Add Note

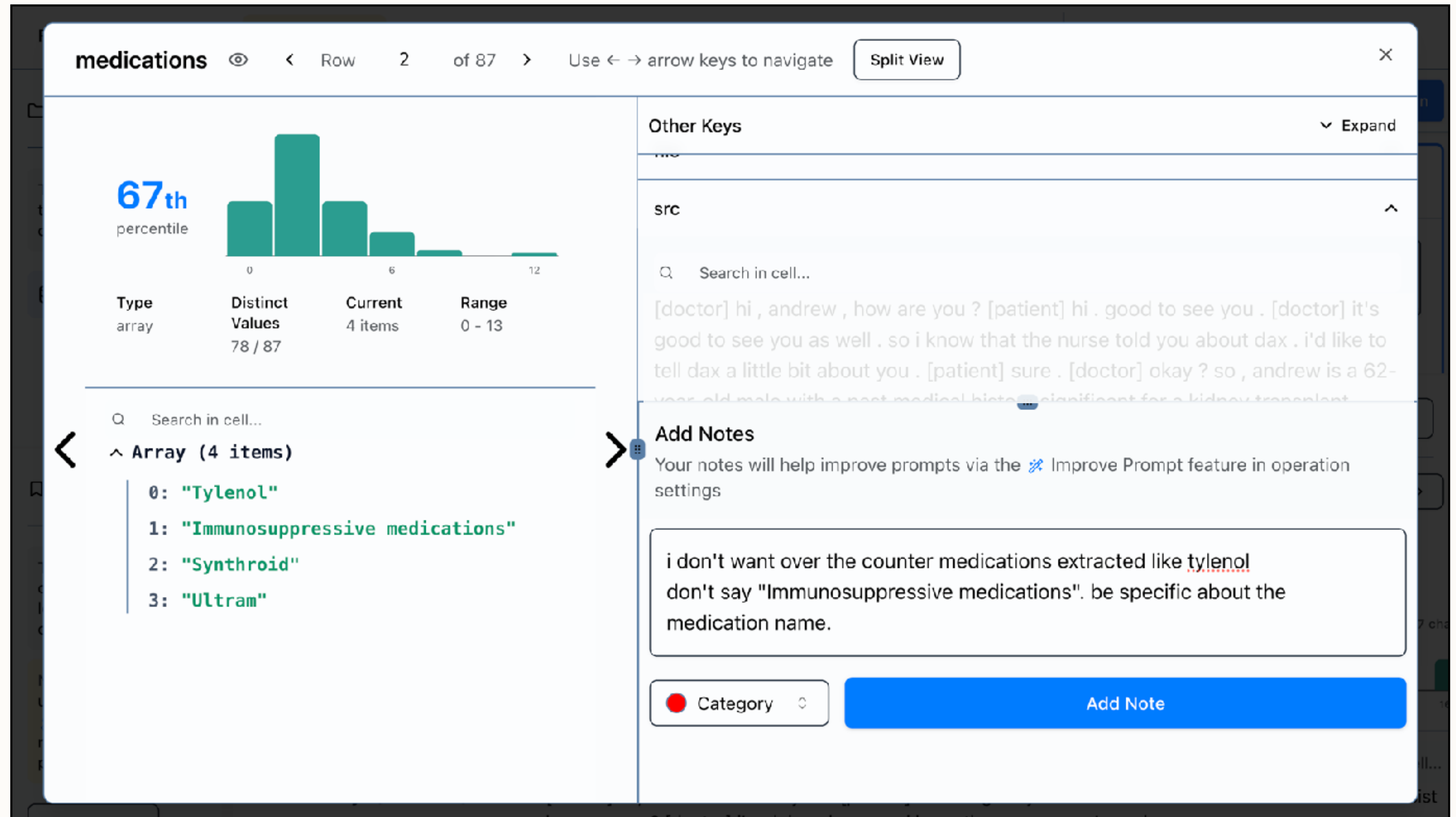
In-Situ Notetaking to *Comprehend* Data Better

Task

"Extract all  from each transcript"

Challenge

Discovering what you really want & the long tail of LLM behaviors



The screenshot shows a data visualization interface for a dataset named "medications". At the top, it indicates "Row 2 of 87" and provides navigation instructions. A "Split View" button is visible. The main area displays a histogram of values, with a "67th percentile" label. Below the histogram, a table summarizes the data: Type (array), Distinct Values (78 / 87), Current (4 items), and Range (0 - 13). A search bar is present. Below the search bar, an "Array (4 items)" is listed: 0: "Tylenol", 1: "Immunosuppressive medications", 2: "Synthroid", and 3: "Ultram". To the right, a panel titled "Other Keys" shows a search bar and a text area containing a transcript snippet. Below this, an "Add Notes" section explains that notes will help improve prompts via the "Improve Prompt" feature. A text input field contains the note: "i don't want over the counter medications extracted like tylenol don't say 'Immunosuppressive medications'. be specific about the medication name." Below the input field, there is a "Category" dropdown menu and a blue "Add Note" button.

Type	Distinct Values	Current	Range
array	78 / 87	4 items	0 - 13

Array (4 items)

- 0: "Tylenol"
- 1: "Immunosuppressive medications"
- 2: "Synthroid"
- 3: "Ultram"

Other Keys

src

Search in cell...

[doctor] hi , andrew , how are you ? [patient] hi . good to see you . [doctor] it's good to see you as well . so i know that the nurse told you about dax . i'd like to tell dax a little bit about you . [patient] sure . [doctor] okay ? so , andrew is a 62-year old male with a past medical history significant for a kidney transplant

Add Notes

Your notes will help improve prompts via the [Improve Prompt](#) feature in operation settings

i don't want over the counter medications extracted like tylenol don't say "Immunosuppressive medications". be specific about the medication name.

Category

Add Note

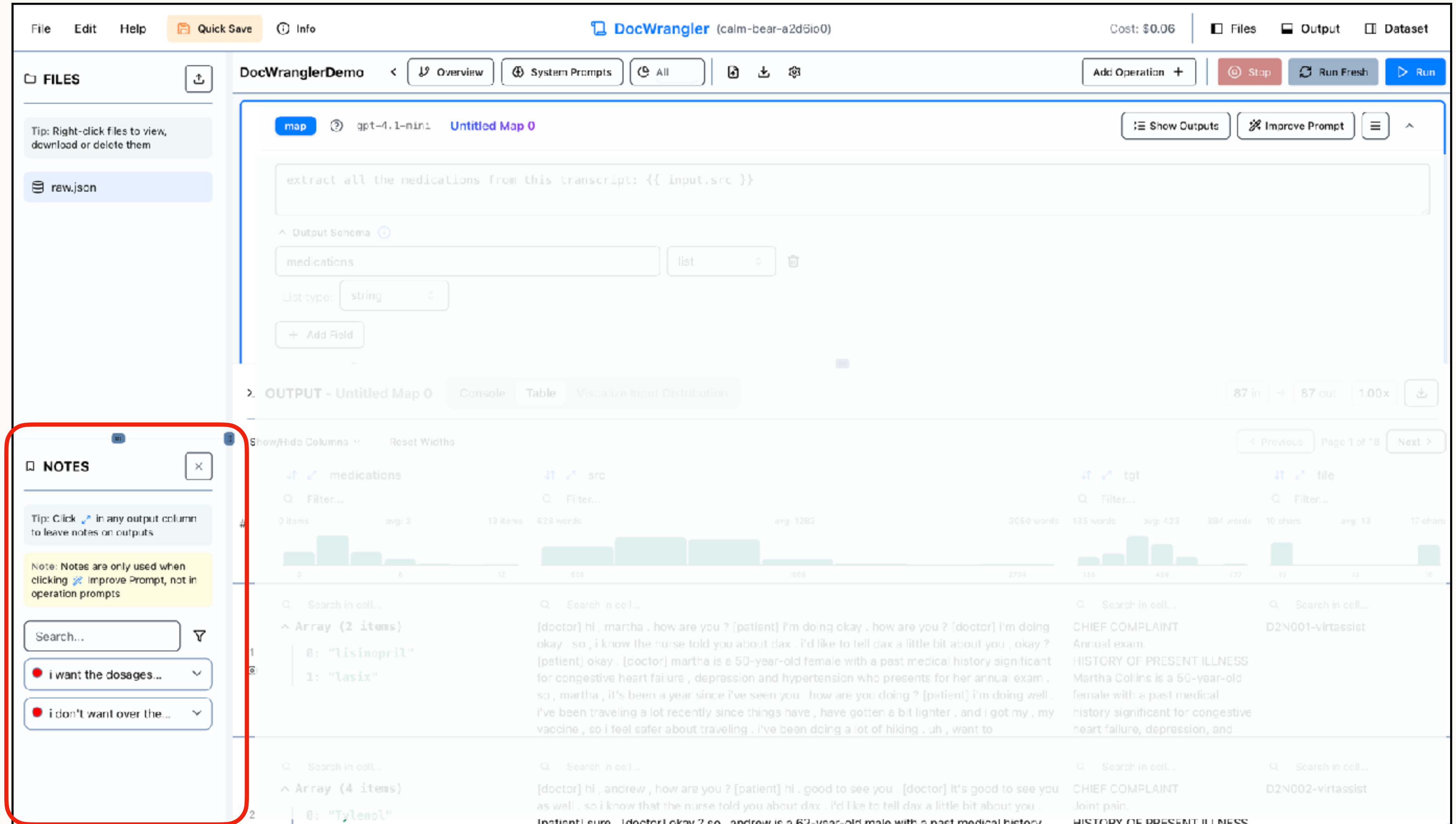
In-Situ Notetaking to *Comprehend* Data Better

Task

"Extract all  from each transcript"

Challenge

Discovering what you really want & the long tail of LLM behaviors



The screenshot displays the DocWrangler interface. At the top, there's a menu bar with 'File', 'Edit', 'Help', 'Quick Save', and 'Info'. The main header shows 'DocWranglerDemo' with navigation tabs for 'Overview', 'System Prompts', and 'All'. A 'Cost: \$0.06' indicator is present. Below the header, a 'FILES' sidebar on the left shows a file named 'raw.json'. The main workspace is titled 'Untitled Map 0' and contains a 'map' operation with a prompt: 'extract all the medications from this transcript: {{ input.src }}'. The 'Output Schema' section shows a field named 'medications' with a 'list' type and a 'string' list type. Below this, the 'OUTPUT - Untitled Map 0' section displays a table view with columns for 'medications', 'src', 'tgt', and 'file'. Each column has a histogram and a search bar. A 'NOTES' sidebar on the left is highlighted with a red box. It contains a tip: 'Tip: Click in any output column to leave notes on outputs.' and a note: 'Note: Notes are only used when clicking Improve Prompt, not in operation prompts.' Below this, there's a search bar and two note entries: 'i want the dosages...' and 'i don't want over the...'. The main table shows two rows of data. The first row has a 'medications' column with an array of two items: 'lisinopril' and 'lasix'. The second row has a 'medications' column with an array of four items: 'Tylenol', 'Tylenol', 'Tylenol', and 'Tylenol'.

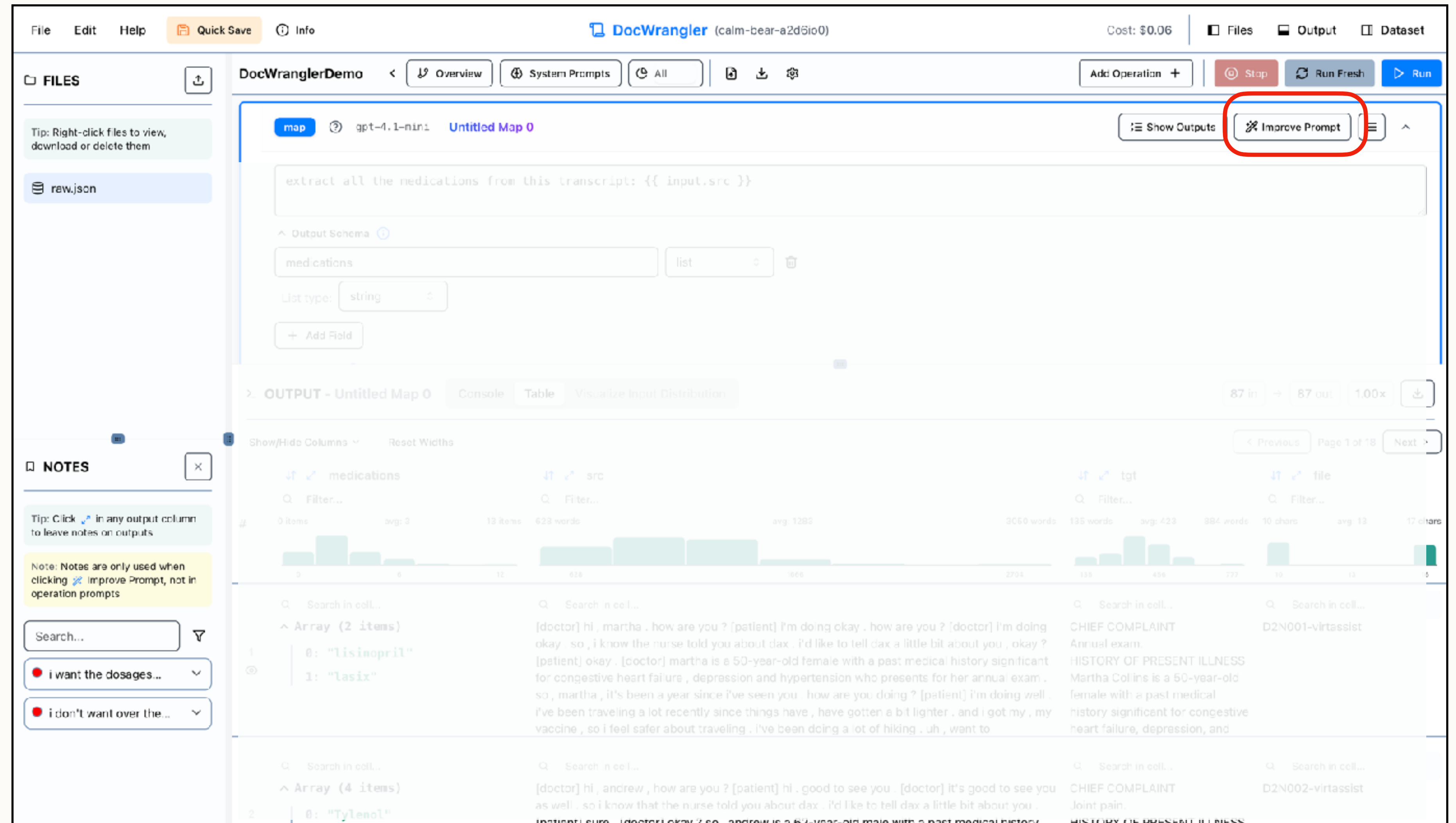
Illustrating the Gulf of Specification

Task

"Extract all  from each transcript"

Challenge

Turning your notes and requirements into better prompts



The screenshot displays the DocWrangler interface. At the top, the title bar shows 'DocWrangler (calm-bear-a2d5io0)' and a cost of '\$0.06'. The main workspace is titled 'DocWranglerDemo' and contains a 'map' operation named 'Untitled Map 0'. The prompt for this operation is 'extract all the medications from this transcript: {{ input.src }}'. The output schema is defined with a field named 'medications' of type 'list', with a 'List type' of 'string'. The output is displayed in a table view, showing two rows of data. The first row contains a list of medications: 'lisinopril' and 'lasix'. The second row contains 'Tylenol'. The interface also includes a 'FILES' sidebar on the left with a file named 'raw.json', and a 'NOTES' sidebar on the right with a search bar and two notes: 'i want the dosages...' and 'i don't want over the...'. A red circle highlights the 'Improve Prompt' button in the top right corner of the map operation's configuration area.

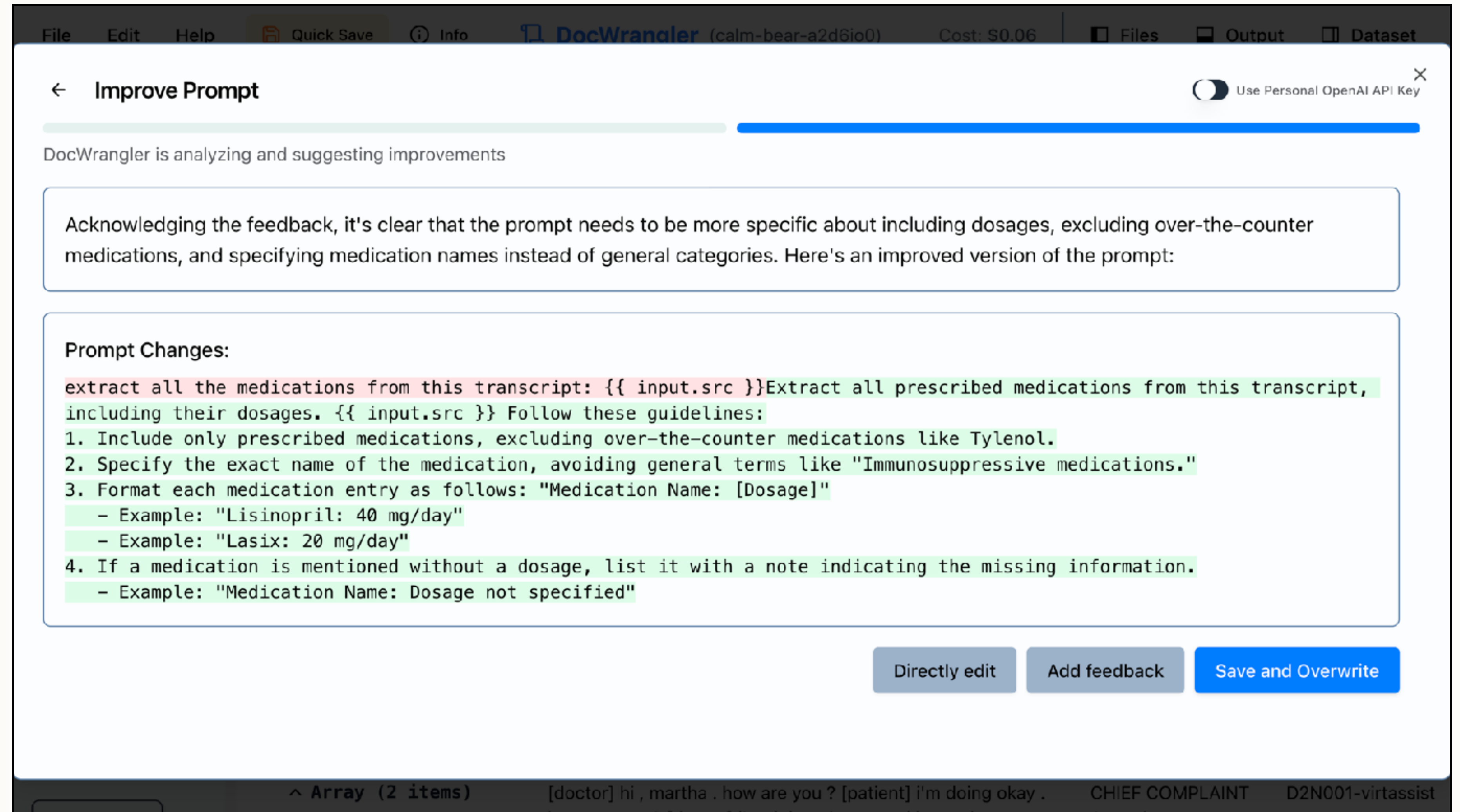
Interactive Assistance in *Specifying* Operations

Task

"Extract all 
from each
transcript"

Challenge

Turning your
notes and
requirements
into better
prompts



The screenshot shows the DocWrangler application interface. At the top, there's a menu bar with 'File', 'Edit', 'Help', 'Quick Save', 'Info', and 'DocWrangler (calm-bear-a2d6io0)'. Below the menu bar, there's a status bar showing 'Cost: \$0.06', 'Files', 'Output', and 'Dataset'. The main content area is titled 'Improve Prompt' and features a progress bar. A message states: 'DocWrangler is analyzing and suggesting improvements'. Below this, a text box explains: 'Acknowledging the feedback, it's clear that the prompt needs to be more specific about including dosages, excluding over-the-counter medications, and specifying medication names instead of general categories. Here's an improved version of the prompt:'. The 'Prompt Changes' section lists four guidelines: 1. Include only prescribed medications, excluding over-the-counter medications like Tylenol. 2. Specify the exact name of the medication, avoiding general terms like 'Immunosuppressive medications.' 3. Format each medication entry as follows: 'Medication Name: [Dosage]' with examples 'Lisinopril: 40 mg/day' and 'Lasix: 20 mg/day'. 4. If a medication is mentioned without a dosage, list it with a note indicating the missing information, with example 'Medication Name: Dosage not specified'. At the bottom right, there are three buttons: 'Directly edit', 'Add feedback', and 'Save and Overwrite'. The bottom status bar shows 'Array (2 items)', '[doctor] hi , martha . how are you ? [patient] i'm doing okay .', 'CHIEF COMPLAINT', and 'D2N001-virtassist'.

File Edit Help Quick Save Info DocWrangler (calm-bear-a2d6io0) Cost: \$0.06 Files Output Dataset

← Improve Prompt Use Personal OpenAI API Key

DocWrangler is analyzing and suggesting improvements

Acknowledging the feedback, it's clear that the prompt needs to be more specific about including dosages, excluding over-the-counter medications, and specifying medication names instead of general categories. Here's an improved version of the prompt:

Prompt Changes:

extract all the medications from this transcript: {{ input.src }}Extract all prescribed medications from this transcript, including their dosages. {{ input.src }} Follow these guidelines:

1. Include only prescribed medications, excluding over-the-counter medications like Tylenol.
2. Specify the exact name of the medication, avoiding general terms like "Immunosuppressive medications."
3. Format each medication entry as follows: "Medication Name: [Dosage]"
 - Example: "Lisinopril: 40 mg/day"
 - Example: "Lasix: 20 mg/day"
4. If a medication is mentioned without a dosage, list it with a note indicating the missing information.
 - Example: "Medication Name: Dosage not specified"

Directly edit Add feedback Save and Overwrite

^ Array (2 items) [doctor] hi , martha . how are you ? [patient] i'm doing okay . CHIEF COMPLAINT D2N001-virtassist

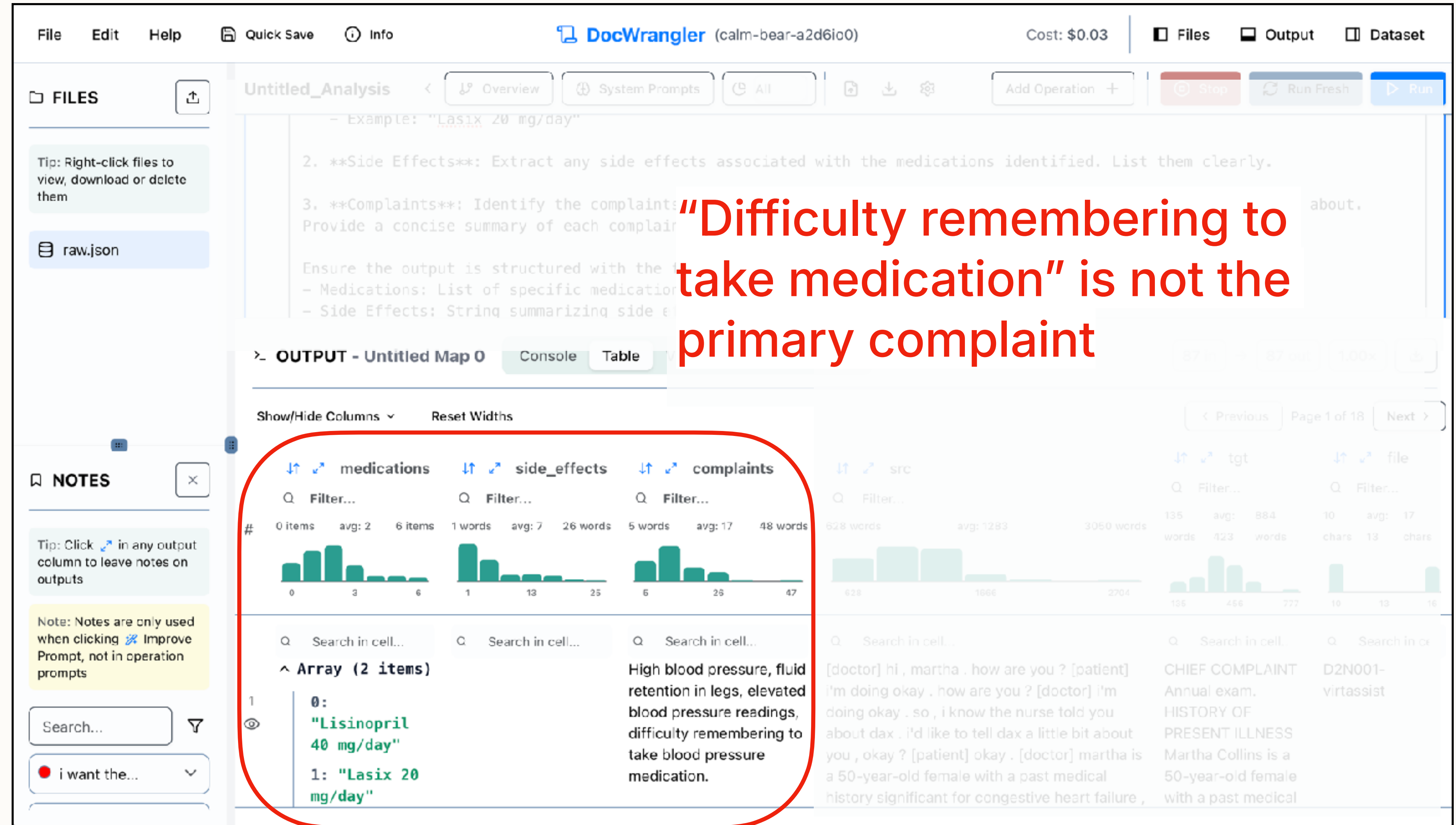
Illustrating the Gulf of Generalization

Task

"Extract all , side effects, and patient complaints from each transcript"

Challenge

Too complex for a single LLM call



DocWrangler (calm-bear-a2d6io0) Cost: \$0.03

File Edit Help Quick Save Info

FILES

Tip: Right-click files to view, download or delete them

raw.json

NOTES

Tip: Click in any output column to leave notes on outputs

Note: Notes are only used when clicking Improve Prompt, not in operation prompts

Search...

i want the...

Untitled_Analysis

Overview System Prompts All

Add Operation + Stop Run Fresh Run

Example: "Lasix 20 mg/day"

2. ****Side Effects****: Extract any side effects associated with the medications identified. List them clearly.

3. ****Complaints****: Identify the complaints. Provide a concise summary of each complaint.

Ensure the output is structured with the following schema:

- Medications: List of specific medication names and dosages.
- Side Effects: String summarizing side effects.
- Complaints: String summarizing patient complaints.

OUTPUT - Untitled Map 0

Console Table

Show/Hide Columns Reset Widths

medications side_effects complaints

Filter... Filter... Filter...

0 items avg: 2 6 items 1 words avg: 7 26 words 5 words avg: 17 48 words

0 3 6 1 13 25 6 26 47

Search in cell... Search in cell... Search in cell...

Array (2 items)

0: "Lisinopril 40 mg/day"

1: "Lasix 20 mg/day"

High blood pressure, fluid retention in legs, elevated blood pressure readings, difficulty remembering to take blood pressure medication.

[doctor] hi , martha . how are you ? [patient] i'm doing okay . how are you ? [doctor] i'm doing okay . so , i know the nurse told you about dax . i'd like to tell dax a little bit about you , okay ? [patient] okay . [doctor] martha is a 50-year-old female with a past medical history significant for congestive heart failure ,

CHIEF COMPLAINT Annual exam.

HISTORY OF PRESENT ILLNESS Martha Collins is a 50-year-old female with a past medical

Page 1 of 18

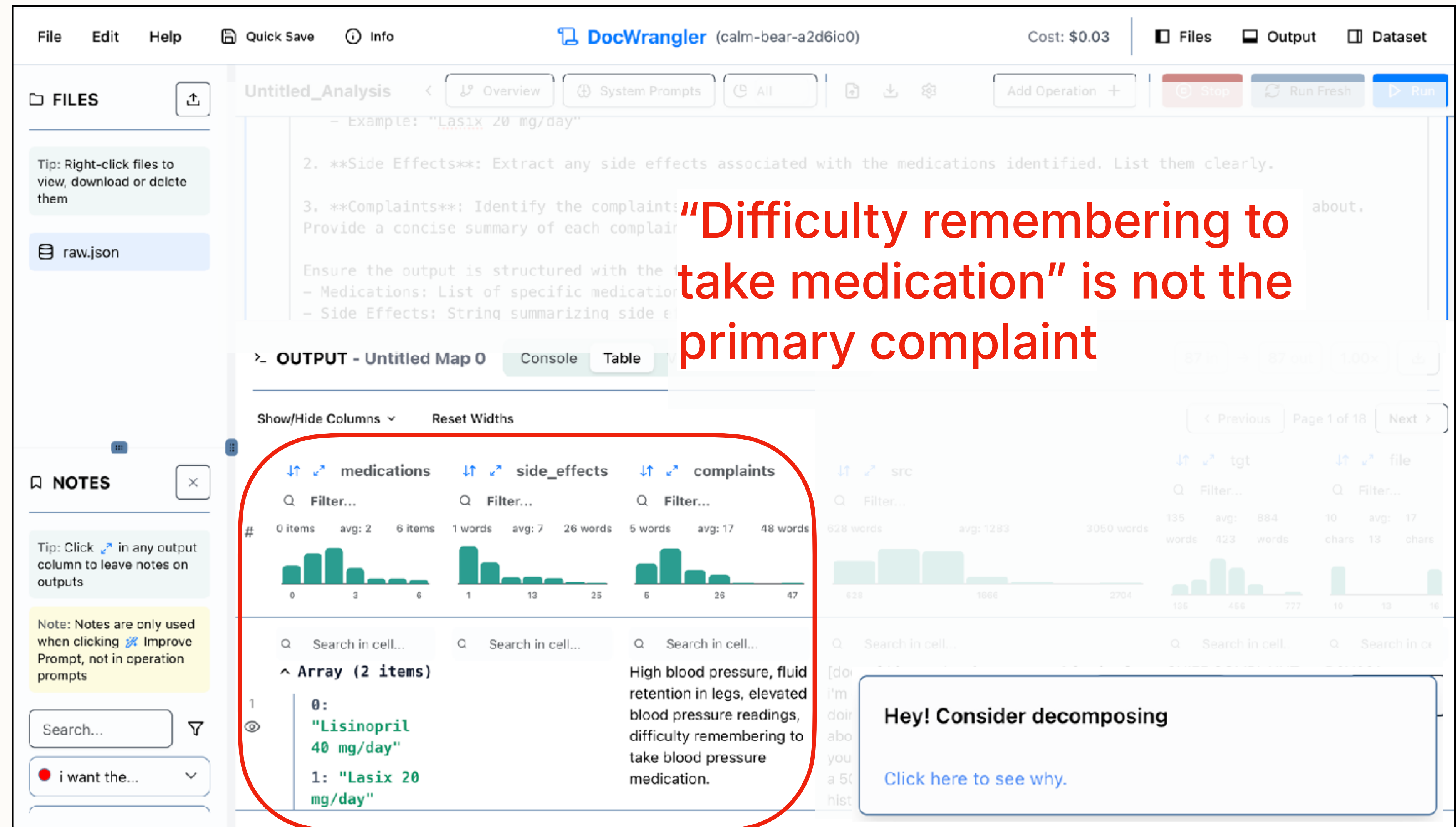
Automatic Decomposition to Help *Generalize*

Task

"Extract all ,
side effects,
and patient
complaints
from each
transcript"

Challenge

Too complex for a single LLM call



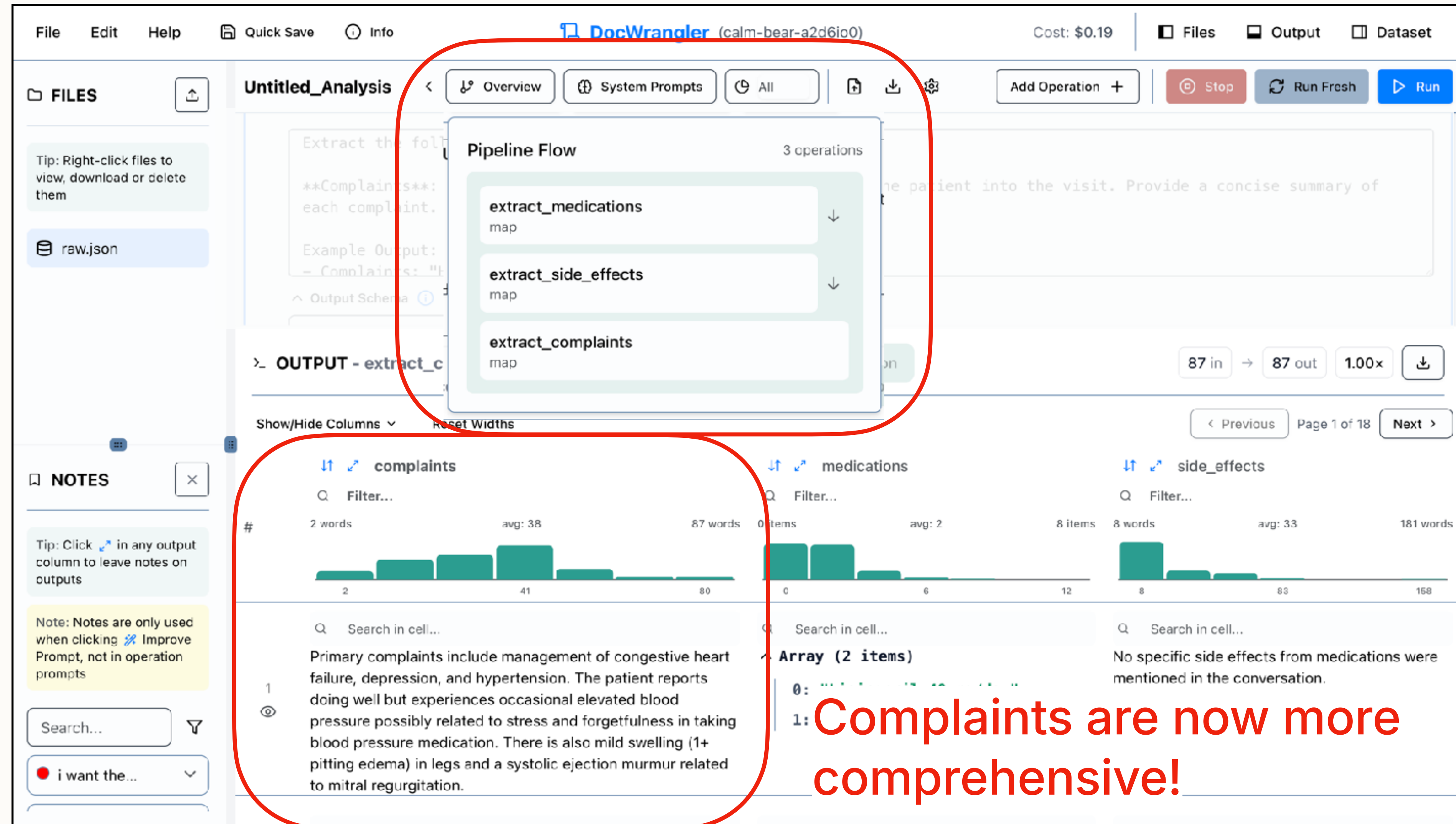
Automatic Decomposition to Help *Generalize*

Task

"Extract all , side effects, and patient complaints from each transcript"

Challenge

Too complex for a single LLM call

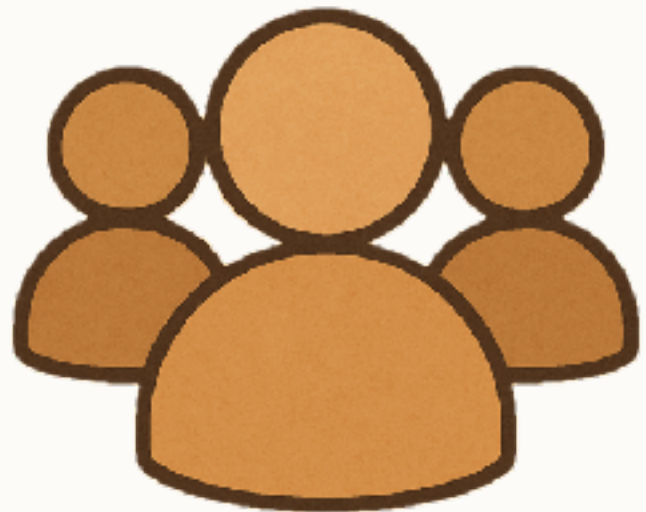


The screenshot displays the DocWrangler interface for an analysis titled "Untitled_Analysis". The interface includes a top navigation bar with "File", "Edit", "Help", "Quick Save", and "Info" options. The main workspace is divided into several sections:

- FILES:** A sidebar on the left showing a file named "raw.json".
- System Prompts:** A section for defining prompts, with a tip: "Tip: Right-click files to view, download or delete them".
- Pipeline Flow:** A central area showing a sequence of three operations: "extract_medications", "extract_side_effects", and "extract_complaints", each labeled as a "map" operation. This section is highlighted with a red circle.
- OUTPUT - extract_c:** A section showing the results of the pipeline. It includes three columns: "complaints", "medications", and "side_effects". Each column has a filter bar and a histogram showing the distribution of results. The "complaints" column shows a distribution with peaks at 2, 41, and 80 words. The "medications" column shows a distribution with peaks at 0, 6, and 12 items. The "side_effects" column shows a distribution with peaks at 8, 83, and 158 words. This section is also highlighted with a red circle.
- NOTES:** A section on the left for adding notes to the output. It includes a tip: "Tip: Click [icon] in any output column to leave notes on outputs".

The output results are displayed in a table format. The "complaints" column shows a single entry: "Primary complaints include management of congestive heart failure, depression, and hypertension. The patient reports doing well but experiences occasional elevated blood pressure possibly related to stress and forgetfulness in taking blood pressure medication. There is also mild swelling (1+ pitting edema) in legs and a systolic ejection murmur related to mitral regurgitation." The "medications" column shows an "Array (2 items)" with two entries: "0: ..." and "1: ...". The "side_effects" column shows a single entry: "No specific side effects from medications were mentioned in the conversation." A red text overlay on the right side of the output section reads: "Complaints are now more comprehensive!"

DocWrangler User Study: Design



10 participant think-aloud lab study

- ◆ 60-90 minute sessions
- ◆ 3-5 analysis tasks per session



Online deployment

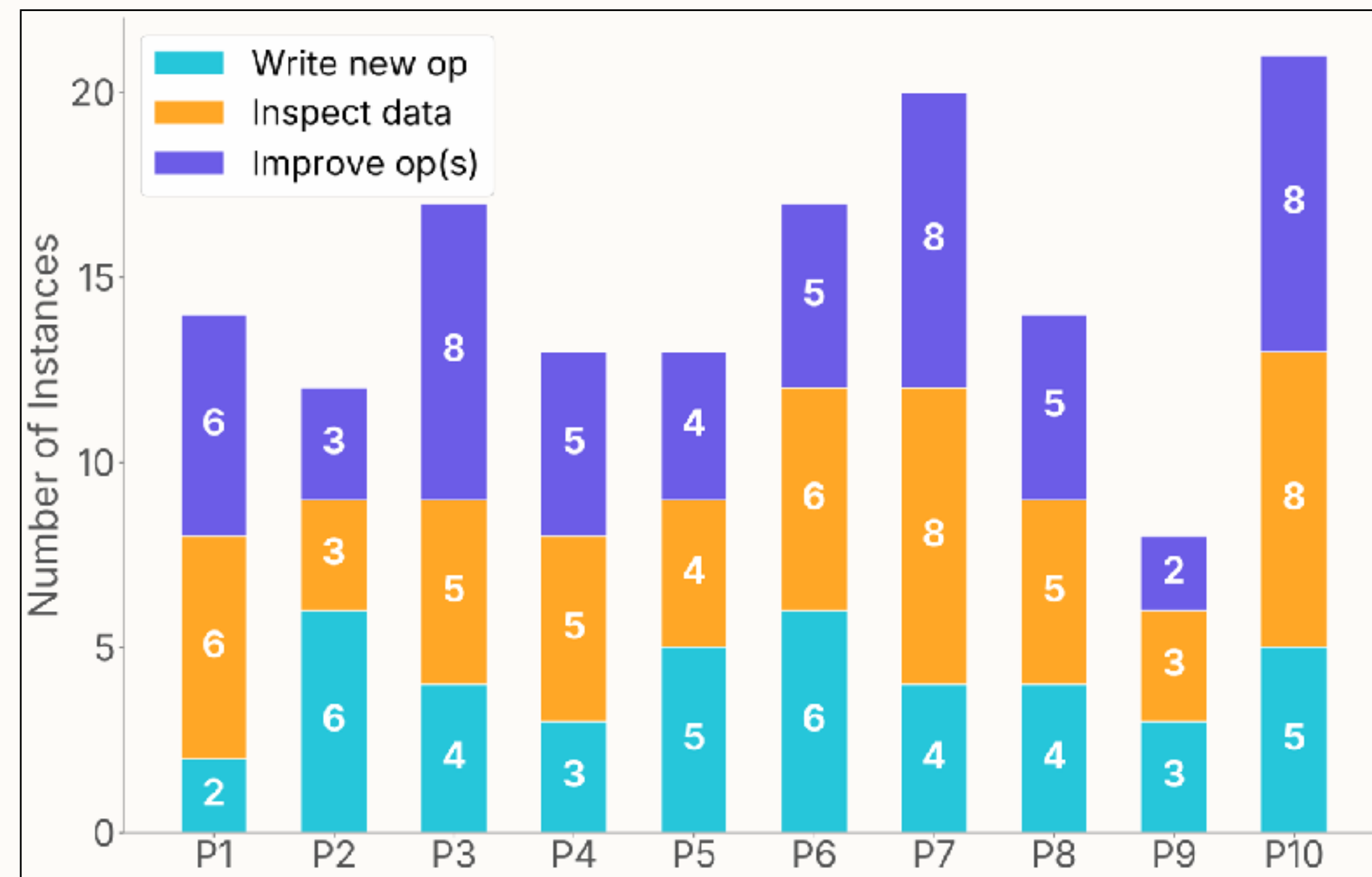
- ◆ BYO API keys
- ◆ ~4,500 pipelines; ~30GB of data (~5 billion words!)
- ◆ 30+ S&P 500 industries represented

Quantitative and qualitative analysis

What Did We Learn?

1. Users iterate *a lot* to get one query

Participant Activity Distribution



What Did We Learn?

1. Users iterate *a lot* to get one query
2. Users don't read raw documents, even when we make it easy

`filter("Does the patient trust the doctor?")`

??

What Did We Learn?

1. Users iterate *a lot* to get one query
2. Users don't read raw documents, even when we make it easy
 - * E.g., they create exploratory ops to *summarize* documents in task-specific ways

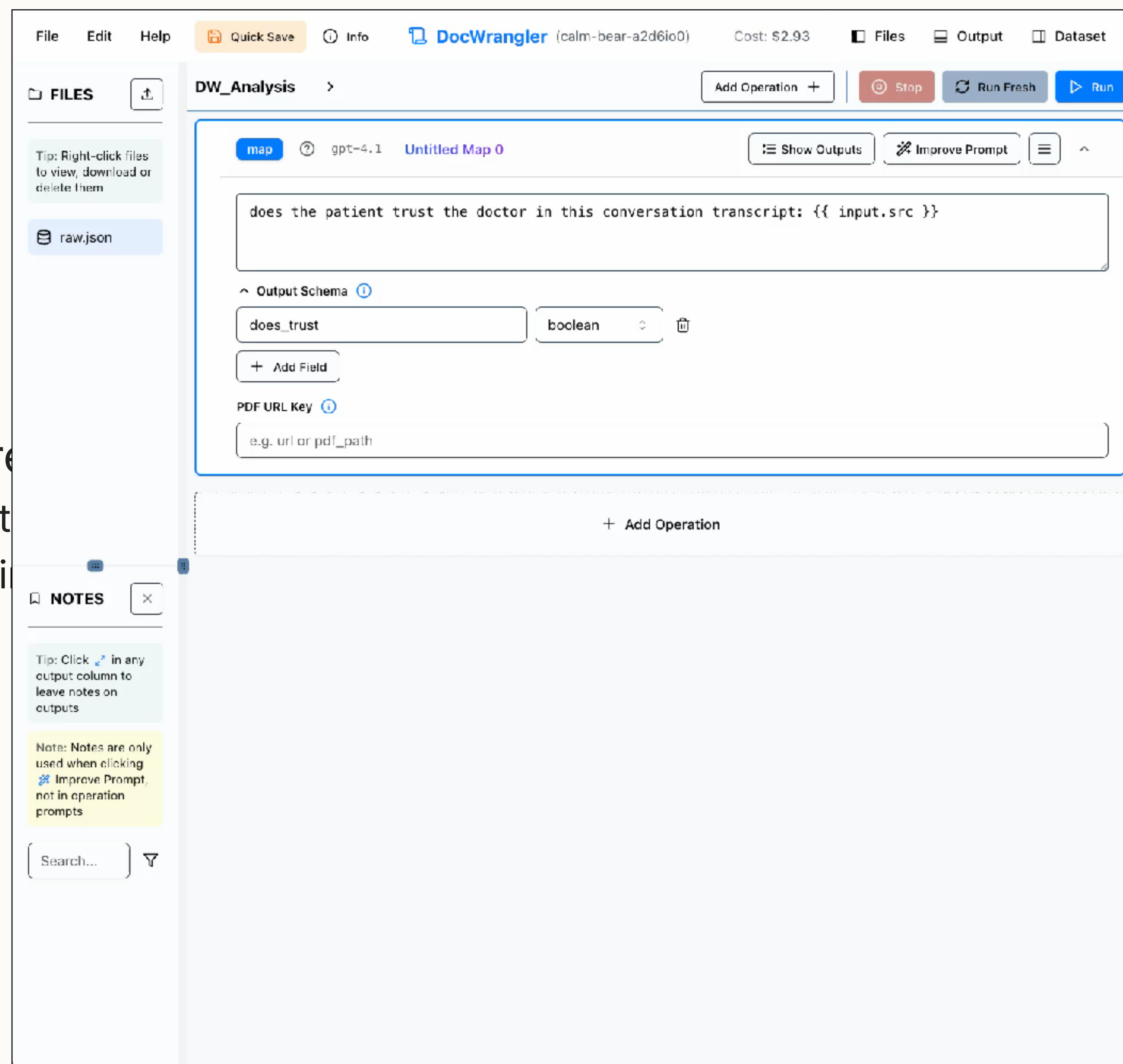
The screenshot shows a query builder interface for the model 'gpt-5-nano'. At the top, there is a blue 'map' button, a help icon, the model name 'gpt-5-nano', and a purple 'summary' button. To the right are buttons for 'Show Outputs', 'Improve Prompt', and a menu icon. Below this is a large text input field containing the prompt 'summarize signs of distrust in {{input.src}}'. Under the input field is an 'Output Schema' section with an expand/collapse arrow and an info icon. It contains a single field named 'summary' of type 'string', with a dropdown arrow and a delete icon. At the bottom left is a '+ Add Field' button.

1. Users iterate

2. Users don't re

- * E.g., they creat

- * Even when tryi



ays

- # 1. Users iterate

- ## 2. Users don't re

- * E.g., they create

- * And open-ended

- ### 3. Underspecific

FileEditHelp

Quick Save

Info

DocWrangler (calm-bear-a2d6io0)

Cost: \$3.32

Files

Output

Dataset

FILES

Tip: Right-click files to view, download or delete them

raw.json

DW_Analysis

Add Operation +

Stop

Run Fresh

Run

map gpt-4.1 Untitled Map 0

Show Outputs

Improve Prompt

does the patient trust the doctor in this conversation transcript: {{ input.src }}

Output Schema

does_trust boolean

reasoning string

+ Add Field

OUTPUT - Untitled Map 0

Console

Table

Visualize Input Distribution

87 in

87 out

1.00x

Show/Hide Columns

Reset Widths

< Previous

Page 1 of 16

Next >

does_trust

Filter...

2 distinct values

true

false

reasoning

Filter...

59 words

avg: 104

113

167

src

Filter...

182 words

628 words

628

true

1

Search in cell...

Throughout the conversation, the patient responds affirmatively and cooperatively to the doctor's questions, advice, and suggestions. The patient voluntarily shares personal and health information, admits difficulties (e.g., forgetting to take medication), honestly details symptoms, and engages in follow-up questions (e.g., about pill timing), indicating openness and willingness to accept the doctor's guidance. She accepts medical recommendations, confirms actions taken, and expresses appreciation at the end ('good seeing you too'). There is no evidence of

true

2

Search in cell...

evident by the patient's cooperative responses ('sure,' 'you got it,' 'perfect'), willingness to answer questions thoroughly, lack of hesitation or resistance to the doctor's suggestions, and no expressed doubt or concern about the doctor's decisions. The patient follows the doctor's lead, responds affirmatively to each part of the care plan, and shows comfort with the examination and assessment process. While the patient does not verbalize explicit trust, the patient's actions, tone, and consistent agreement to the doctor's recommendations (such as taking prescribed

Search...

Tip: Click in any output column to leave notes on outputs

Note: Notes are only used when clicking Improve Prompt, not in operation prompts

What Did We Learn?

1. Users iterate *a lot* to get one query
2. Users don't read raw documents, even when we make it easy
 - * E.g., they create exploratory ops to *summarize* documents in task-specific ways
 - * And open-ended exploratory *attributes* to validate LLM behavior
3. Underspecification drives discovery
 - * E.g., Users tune attribute granularity & use visualizations to refine
 - * "Prompt rubber ducking" (P9) — outputs help users discover more things to ask for

What Did We Learn?

1. Users iterate *a lot* to get one query
2. Users don't read raw documents, even when we make it easy
 - * E.g., they create exploratory ops to *summarize* documents in task-specific ways
 - * And open-ended exploratory *attributes* to validate LLM behavior
3. Underspecification drives discovery
 - * E.g., Users tune attribute granularity & use visualizations to refine
 - * "Prompt rubber ducking" (P9) — outputs help users discover more things to ask for
4. Users need to slow down!

What Did We Learn?

1. Users iterate *a lot* to get one query
2. Users don't read raw documents, even when we make it easy
 - * E.g., they create exploratory ops to *summarize* documents in task-specific ways
 - * And open-ended exploratory *attributes* to validate LLM behavior
3. Underspecification drives discovery
 - * E.g., Users tune attribute granularity & use visualizations to refine
 - * "Prompt rubber ducking" (P9) — outputs help users discover more things to ask for
4. Users need to slow down!
 - * >50% of prompts auto-refined with *zero notes* 🤨
 - * Output schemas as "speed brakes" (P8) — forces users to think about what they want

Impact: DocWrangler Users



Impact: DocETL Community

ucbepic / docetl

<> Code

Issues 28

Pull requests

👁

🔗

★

▼

A system for agentic LLM-powered data processing and ETL

[docetl.org](#)

MIT license

3.1k stars 320 forks 26 watching

112 Branches 15 Tags Activity

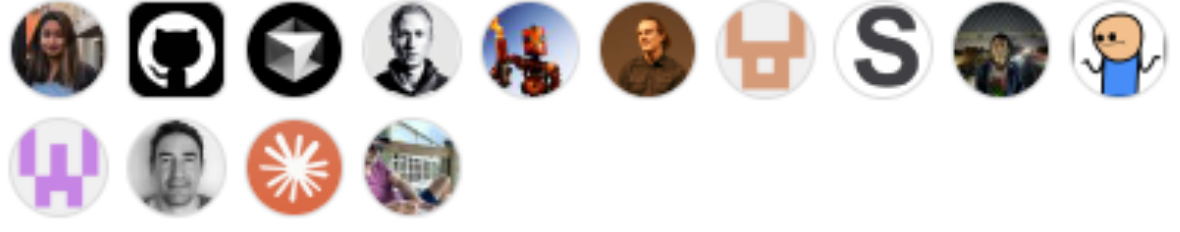
docetl

36 Online 472 Members

Boost

Invite

Contributors 28



+ 14 contributors

DocETL - AI-Powered Docu...

Claude

[2410.12188] DocETL: Agentic...

localhost:3100/playground

File Edit Help Quick Save Info

DocWrangler (blog-post-llm-context) Cost: \$0.00

FILES

Tip: Right-click files to view, download or delete them

converted: 2025-09-17T16-02-59-013Z

2021-04-12-fastai-chapter-6.md

2021-04-25-fastai-chapter-6-bear-classifier.md

NOTES

Tip: Click ⌘ in any ... n outputs

Note: Notes are onl... n prompts

Note: Notes are onl... n prompts

Search...

The chunk is small enough where the takeaways are pretty...

No explanatory text is in this chunk, so the LLM has inferred...

The document title is not included in this set of key points...

The key points are valid and they match the Markdown chunk...

The key points here are unnecessary and somewhat...

I want document title removed from the LLM prompt.

Blog Post LLM Context

Tip: Right-click files to view, download or delete them

Please follow these steps:

Summarize Information: Condense the extracted information from this section into key points.

Output Format:

The final output should be structured as follows:

Section Key Points:

Summary of key point 1

Summary of key point 2

Summary of key point 3

Example Output:

Document Title: "Market Analysis 2023"

Key Points:

Point 1: "Market growth projected at 10% YoY."

Point 2: "Emerging competitors include Company A and Company B."

Point 3: "Consumer preferences shifting towards eco-friendly products."

OUTPUT - Untitled Map 0

Console Table Visualize Input Distribution

key_points filename markdown

Filter...

3 items avg: 4 6 items 5 distinct values

Search in cell...

rather than a discrete category

2021-04-12-fastai-chapter-6

title: "fast.ai Chapter 6: Classi...

which I think is pretty cool, is

for these classification

Using DocWrangler to Process Blog Posts

vishal 1.82K subscribers

Subscribe

@EkShunya 1 month ago

nice to see you back, docwrangler +1

👍 🔄 🗨 Reply

@alexkelly757 1 month ago

I was wondering about resolve myself. decided to go for reduce. It was interesting your thoughts process on map and reduce.

👍 🔄 🗨 Reply

@vishalJleamer 1 month ago

Thanks I'll try out gpt5 nano for sure. I need to experiment with more models. Regarding the final map: the reason I'm aggregating across multiple documents, it was summarizing within a single document. But I'm sure conceptually it's

👍 1 🗨 Reply

Solvelt + FastHTML: Creating Interactive Search Applications

AI HELP

AUTHOR

Alex Kelly

PUBLISHED

October 4, 2025

Displaying enriched docETL data using solvit/fasthtml

The reason for this project is I have data that is currently in Excel, I want to build a front end website to display the information and use advanced information retrievable. The data was enriched using DocETL which enables you to build a series of LLM operations to extract and enrich information from unstructured data. I see these LLM operations been key in every organization and being able to display and manipulate the data as a key skill. In this case it is a fictitious set of email data. Lets dive into the code :

Building a FastHTML + LanceDB Search App Using Solvelt (Jeremy H...)

solvelt Testing fasthtml join final out

Watch on YouTube

imports and setups

```
import LanceDb
import numpy as np
from LanceDb.pydantic import LanceModel, Vector
from LanceDb.embeddings import get_registry
import sys
```

What Do Users Actually Do with LLM-Powered Data Systems?

Shreya Shankar and Aditya G. Parameswaran
UC Berkeley
{shreyashankar, adityagp}@berkeley.edu

Abstract

Large language models (LLMs) are now widely integrated into many database management systems. Users can now define relational operators in natural language and query unstructured data at scale, with the database using LLMs under the hood. Our community has devised many techniques to make such LLM-powered queries cheaper and more accurate, but all of our work assumes the user arrives with a well-specified query and ground-truth labels (or a labeling function). Through building and deploying DocETL, an open-source LLM-powered data analysis system, and DocWrangler, an IDE on top of it, we found that this assumption rarely holds. Drawing on 1,173 queries from an 8-month deployment and a long-term collaboration with public defenders on the California Racial Justice Act, we examine how users actually use LLM-powered data systems: how they struggle to specify and validate queries, how they iterate extensively to converge on bespoke operators specific to their documents and domains, and what these patterns imply for the systems and benchmarks our community should be building.

1 Introduction

Large language models (LLMs) have upended data management. For the first time, users can query unstructured data like long-form documents and images right inside their databases, at scale and in genuinely complex ways. Many major data systems now let users define relational operators (e.g., filter, map, join) in natural language and have the database invoke LLMs to execute them. For example, Snowflake Cortex exposes operators like `AI_FILTER`, `AI_CLASSIFY`, `AI_COMPLETE`, `AI_JOIN`, and `AI_AGG` [1, 2]; BigQuery offers `AI.GENERATE_TABLE` and `ML.GENERATE_TEXT` [3]; Google AlloyDB provides a similar suite [4]. We will refer to the interface as “AI-X”—e.g., AI-SQL, AI-MapReduce, AI-Pandas—from hereon out.

The broad goal of AI-X is to let users *simply* specify what they want in natural language, declaratively, and leave all choices about how to execute the query to the database. For example, suppose a user has a table of customer reviews, with schema `reviews(id, review_txt)`, and wants to find the reviews from upset customers. An AI-SQL query for this might look like `SELECT * FROM reviews WHERE`

Year Later



**IEEE Data
Engineering
Bulletin; June 2026**

and reusable (templates) often
c oper
(Lee et al. VLDB 2025)

Summary

Unstructured & LLM-powered data systems: a new subfield.

New usability challenges for data systems; in

Comprehending the documents

Specifying intent in natural language prompts

And ensuring that LLMs are **generalizing** to all documents.

What's next? I'm starting a new lab and hiring! We will build open-source data systems & interfaces for human-AI collaboration.

